Spectral computation with third-order tensors using the t-product

Anas El Hachimi[b,a], Khalide Jbilou[b,a], Ahmed Ratnani[a], Lothar Reichel[c]

[a]*The UM6P Vanguard Center, Mohammed VI Polytechnic University, Green City, Morocco.*
[b]*Université du Littoral Cote d'Opale, LMPA, 50 rue F. Buisson, 62228 Calais-Cedex, France.*
[c]*Department of Mathematical Sciences, Kent State University, Kent, OH 44242, USA.*

Abstract

The tensor t-product is a powerful tool for the analysis of and computation with third-order tensors. This paper discusses properties and the computation of eigentubes and eigenslices of third-order tensors under the t-product; the eigentubes and eigenslices are analogues of eigenvalues and eigenvectors for matrices. The computational methods considered and analysed include the tensor power method, tensor subspace iteration, and the tensor QR algorithm. Computed examples illustrate the performance of these methods.

Keywords: tensor, t-product, eigentube, eigenslice, tensor power method, tensor subspace iteration, tensor QR algorithm

1. Introduction

Tensors are high-dimensional generalizations of matrices. They find many applications in science and engineering, including in image processing, data mining, computer vision, network analysis, and the solution of partial differential equations; see [2, 3, 4, 5, 6, 7, 8, 13, 19, 21, 31] and references therein. Several different tensor products have been proposed in the literature. The n-mode product is one of the most commonly used products; it defines the product of a tensor and a matrix of appropriate sizes; see [3, 14]. The CP and Tucker tensor decompositions use this product; see [3, 9]. The oldest tensor product is probably the Einstein product [11]; it has recently been applied to color image and video processing [17]. The tensor t-product for third-order tensors used in the paper was proposed by Kilmer et al. in [26, 27]. This product has found many applications and has been generalized in several ways; see [1, 2, 6, 7, 15, 16, 18, 20, 29, 31, 32]. The t-product allows natural generalizations of many

Email addresses: `anaselhachimi1997@gmail.com` (Anas El Hachimi), `khalide.jbilou@univ-littoral.fr` (Khalide Jbilou), `ahmed.ratnani@um6p.ma` (Ahmed Ratnani), `reichel@math.kent.edu` (Lothar Reichel)

matrix factorizations including the singular value decomposition, and the QR and Choleski factorizations to third-order tensors; see [26, 27, 33].

Several notions of eigenvalues for a tensor have been described in the literature. They include the C-eigenvalues, H-eigenvalues, and Z-eigenvalues; see [12, 30]. The definitions depend on the tensor product used. The eigenvalues in these definitions are scalars. We will use the t-product and discuss properties of eigentubes and eigenslices, which are analogues for third-order tensors of eigenvalues and eigenvectors, respectively, for matrices. Moreover, we will describe algorithms for the computation of eigentubes and eigenslices.

The organization of this paper is as follows: Sections 2 reviews properties of the t-product, Section 3 describes some tensor decompositions, and Section 4 discusses some properties of eigentubes and eigenslices. Numerical methods for the computation of a few or all eigentubes and associated eigenslices of a tensor are described in Section 5. A few computed examples are presented in Section 6, and Section 7 contains concluding remarks.

2. The t-product for third order tensors

This section reviews definitions and results by Kilmer et al. [26, 27] and uses notation from there, as well as from Kolda and Bader [3]. Consider the third-order tensor $\mathscr{A} = [\mathscr{A}_{i,j,k}] \in \mathbb{C}^{\ell \times p \times n}$. A *slice* of $\mathscr{A}$ is a 2D section obtained by fixing any one of the indices. Using MATLAB-type notation, $\mathscr{A}_{i,:,:}$, $\mathscr{A}_{:,j,:}$, and $\mathscr{A}_{:,:,k}$ stand for the ith horizontal, jth lateral, and kth frontal slices of $\mathscr{A}$, respectively. We also denote the jth lateral slice by $\vec{\mathscr{A}_j}$, which is a tensor. Sometimes it is convenient to identify $\vec{\mathscr{A}_j}$ with a matrix. The kth frontal slice, $\mathscr{A}_{:,:,k}$ is also denoted by $\mathscr{A}^{(k)}$ and is a matrix. A *fiber* of a third order tensor $\mathscr{A}$ is a 1D section obtained by fixing any two of the indices. Thus, $\mathscr{A}_{:,j,k}$, $\mathscr{A}_{i,:,k}$, and $\mathscr{A}_{i,j,:}$ denote mode-1, mode-2, and mode-3 fibers (tubes), respectively. We denote the set of $\ell \times p \times n$ complex third-order tensors by $\mathbb{K}_n^{\ell \times p}$. The set $\mathbb{K}_n^{\ell}$ stands for complex tensors of size $\ell \times 1 \times n$ and is referred to as the space of lateral slices, and $\mathbb{K}_n$ denotes the space of complex tubes, i.e., elements of size $1 \times 1 \times n$. Many of the tensors considered will be *square*, i.e., $\ell = p$.

The inner product of two third-order tensors $\mathscr{A}, \mathscr{B} \in \mathbb{K}_n^{\ell \times p}$ is given by

$$\langle \mathscr{A}, \mathscr{B} \rangle = \sum_{i,j,k=1}^{\ell,p,n} \mathscr{A}_{i,j,k} \bar{\mathscr{B}}_{i,j,k},$$

where the bar denotes complex conjugation, and the Frobenius norm of $\mathscr{A}$ is defined as

$$\|\mathscr{A}\|_F = \langle \mathscr{A}, \mathscr{A} \rangle^{1/2}.$$

We note that a third-order tensor $\mathscr{A} \in \mathbb{K}_n^{\ell \times p}$ can be represented as

$$\mathscr{A} = \left[\vec{\mathscr{A}_1}, \vec{\mathscr{A}_2}, \ldots, \vec{\mathscr{A}_p} \right].$$

The Discrete Fourier Transform (DFT) is helpful for the efficient evaluation of the t-product (defined below) of two tensors. The DFT of a vector $v \in \mathbb{C}^n$ is given by

$$\widehat{v} = F_n v \in \mathbb{C}^n, \tag{1}$$

where F_n denotes the DFT matrix defined as

$$F_n = \left[\left(e^{\frac{2i\pi}{n}} \right)^{(k-1)(j-1)} \right]_{k=1:n, j=1:n} \in \mathbb{C}^{n \times n}.$$

While the matrix F_n is not unitary, the scaled matrix $F_n/\sqrt{n}$ is. The computation of the vector $\widehat{v}$ by straightforward evaluation of the matrix-vector product in the right-hand side of (1) requires $O(n^2)$ arithmetic floating point operations (flops). It is well known that this flop count is reduced to $O(n\log(n))$ when using the fast Fourier transform (FFT). This can be done in MATLAB with the command $\widehat{v} = \texttt{fft}(v)$.

Let $\widehat{\mathscr{A}} \in \mathbb{K}_n^{\ell \times p}$ denote the tensor obtained by applying the DFT along the third dimension of the tensor $\mathscr{A} \in \mathbb{K}_n^{\ell \times p}$. Thus, we evaluate the DFT of each tube of $\mathscr{A}$; this can be done in MATLAB with the command

$$\widehat{\mathscr{A}} = \texttt{fft}\,(\mathscr{A}, [\,], 3)\,.$$

To transform $\widehat{\mathscr{A}}$ back to $\mathscr{A}$, we use the MATLAB command

$$\mathscr{A} = \texttt{ifft}\left(\widehat{\mathscr{A}}, [\,], 3\right).$$

For $\mathscr{A}, \mathscr{B} \in \mathbb{K}_n^{\ell \times p}$, we have

$$\|\mathscr{A}\|_F = \frac{1}{\sqrt{n}} \|\widehat{\mathscr{A}}\|_F, \quad \langle \mathscr{A}, \mathscr{B} \rangle = \frac{1}{n} \langle \widehat{\mathscr{A}}, \widehat{\mathscr{B}} \rangle,$$

where $\widehat{\mathscr{B}} = \texttt{fft}\,(\mathscr{B}, [\,], 3)$.

The following definitions are required below. For $\mathscr{A} \in \mathbb{K}_n^{\ell \times p}$, we define the associated block diagonal matrix

$$\texttt{bdiag}\,(\mathscr{A}) = \begin{pmatrix} \mathscr{A}^{(1)} & & & & \\ & \mathscr{A}^{(2)} & & & \\ & & \mathscr{A}^{(3)} & & \\ & & & \ddots & \\ & & & & \mathscr{A}^{(n)} \end{pmatrix} \in \mathbb{C}^{\ell n \times pn}.$$

Also introduce the block circulant matrix

$$\texttt{bcirc}\,(\mathscr{A}) = \begin{pmatrix} \mathscr{A}^{(1)} & \mathscr{A}^{(n)} & \mathscr{A}^{(n-1)} & \dots & \mathscr{A}^{(2)} \\ \mathscr{A}^{(2)} & \mathscr{A}^{(1)} & \mathscr{A}^{(n)} & \dots & \mathscr{A}^{(3)} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathscr{A}^{(n-1)} & \mathscr{A}^{(n-2)} & \dots & \mathscr{A}^{(1)} & \mathscr{A}^{(n)} \\ \mathscr{A}^{(n)} & \mathscr{A}^{(n-1)} & \dots & \mathscr{A}^{(2)} & \mathscr{A}^{(1)} \end{pmatrix} \in \mathbb{C}^{\ell n \times pn}.$$

The operators unfold and fold are defined as

$$\texttt{unfold}\,(\mathscr{A}) = \begin{pmatrix} \mathscr{A}^{(1)} \\ \mathscr{A}^{(2)} \\ \vdots \\ \mathscr{A}^{(n)} \end{pmatrix} \in \mathbb{C}^{\ell n \times p} \quad \text{and} \quad \texttt{fold}(\texttt{unfold}\,(\mathscr{A})) = \mathscr{A}.$$

Thus, unfold transforms the tensor $\mathscr{A} \in \mathbb{K}_n^{p \times \ell}$ into a matrix and fold is the inverse operator.

A block circulant matrix can be block diagonalized by using the DFT. Kilmer et al. [26] use this property to establish that for $\mathscr{A} \in \mathbb{K}_n^{\ell \times p}$, we have

$$(F_n \otimes I_\ell)\, \texttt{bcirc}\,(\mathscr{A})\, \left(F_n^H \otimes I_p\right) = \texttt{bdiag}(\widehat{\mathscr{A}}),$$

where $\otimes$ stands for the Kronecker product. The following definition is due to Kilmer et al. [26].

Definition 1. *Let $\mathscr{A} \in \mathbb{K}_n^{\ell \times q}$ and $\mathscr{B} \in \mathbb{K}_n^{q \times p}$. Then the t-product of $\mathscr{A}$ and $\mathscr{B}$ is defined as*

$$\mathscr{A} * \mathscr{B} = \texttt{fold}\,(\texttt{bcirc}\,(\mathscr{A})\, \texttt{unfold}\,(\mathscr{B})) \in \mathbb{K}_n^{\ell \times p}. \tag{2}$$

The evaluation of the t-product by using the above definition is quite costly unless the tensors $\mathscr{A}$ and $\mathscr{B}$ are very sparse. Kilmer et al. [26] therefore propose to evaluate the t-product (2) with the aid of the FFT. The following result makes this possible.

Proposition 1. *([26]) Let $\mathscr{A} \in \mathbb{K}_n^{\ell \times q}$ and $\mathscr{B} \in \mathbb{K}_n^{q \times p}$. Then the tensor $\mathscr{C} = \mathscr{A} * \mathscr{B}$ can be determined from*

$$\texttt{bdiag}(\widehat{\mathscr{C}}) = \texttt{bdiag}(\widehat{\mathscr{A}})\, \texttt{bdiag}(\widehat{\mathscr{B}}).$$

We briefly consider the special case of evaluating the t-product of real tensors $\mathscr{A} \in \mathbb{R}^{\ell \times q \times n}$ and $\mathscr{B} \in \mathbb{R}^{q \times p \times n}$. It is well known that for $v \in \mathbb{R}^n$, the vector $\widehat{v} = [\widehat{v}_1, \ldots, \widehat{v}_n$ satisfies

$$\widehat{v}_1 \in \mathbb{R} \quad \text{and} \quad \bar{\widehat{v}}_i = \widehat{v}_{n-i+2} \quad \text{for} \quad i = 2, \ldots, \left[\frac{n+1}{2}\right], \tag{3}$$

where $[(n+1)/2]$ stands for the integer part of $(n+1)/2$. A vector that satisfies (3) is said to be conjugate-even. Moreover, if the n-vector $\tilde{v}$ is real and conjugate-even, then $v = F_n^H \tilde{v}$ is a real conjugate-even vector, where the superscript ^{H} denotes transposition and complex conjugation; see, e.g., [24, Chapter 15] for many properties of the DFT, as well as [34].

Let the operator vec denote the vectorization of a tube. We say that the tube $\boldsymbol{a} \in \mathbb{R}^{1 \times 1 \times n}$ is real conjugate-even, if the vector $a = \texttt{vec}(\boldsymbol{a})$ is real conjugate-even. By exploiting the symmetry of the DFT, it follows that for real third-order tensors $\mathscr{A} \in \mathbb{R}^{\ell \times p \times n}$, we have

$$\widehat{\mathscr{A}}^{(1)} \in \mathbb{R}^{\ell \times p} \quad \text{and} \quad \overline{\widehat{\mathscr{A}}^{(i)}} = \widehat{\mathscr{A}}^{(n-i+2)},\ i = 2, 3, \ldots, \left[\frac{n-1}{2}\right].$$

Algorithm 1 exploits the symmetry properties of the DFT when evaluating the t-product of real tensors.

Algorithm 1 t-product of real third-order tensors

```
Input: $\mathscr{A} \in \mathbb{R}^{\ell \times q \times n}$, $\mathscr{B} \in \mathbb{R}^{q \times p \times n}$.
Output: $\mathscr{C} = \mathscr{A} * \mathscr{B} \in \mathbb{R}^{\ell \times p \times n}$.
Compute $\widehat{\mathscr{A}} = \texttt{fft}(\mathscr{A}, [\,], 3)$ and $\widehat{\mathscr{B}} = \texttt{fft}(\mathscr{B}, [\,], 3)$
for $i = 1, 2$ to $\left[\frac{n+1}{2}\right]$ do
    $\widehat{\mathscr{C}}^{(i)} = \widehat{\mathscr{A}}^{(i)} \widehat{\mathscr{B}}^{(i)}$
end for
for $i = \left[\frac{n+1}{2}\right] + 1$ to $n$ do
    $\widehat{\mathscr{C}}^{(i)} = \overline{\widehat{\mathscr{C}}^{(n-i+2)}}$
end for
Compute $\mathscr{C} = \texttt{ifft}\left(\widehat{\mathscr{C}}, [\,], 3\right)$
```

We list some definitions and the main properties of the t-product that will be used in the sequel. Proofs and most of these definitions can be found in [26].

Proposition 2. *(i) The identity tensor under the t-product, $\mathscr{I}_\ell \in \mathbb{K}_n^{\ell \times \ell}$, has the first frontal slice equal to the identity matrix and the remaining frontal slices are zero matrices.*

(ii) The canonical lateral slice under the t-product, $\vec{\mathscr{E}}_j \in \mathbb{K}_n^{\ell}$, where $1 \leqslant j \leqslant \ell$, has the $(j, 1, 1)$th entry equal to one, and all other entries zero.

(iii) The unit tube, $\boldsymbol{e} \in \mathbb{K}_n$, has the entry $(1, 1, 1)$ equal to one, and all other entries zero.

(iv) The conjugate transpose, $\mathscr{A}^H \in \mathbb{K}_n^{p \times \ell}$, of the tensor $\mathscr{A} \in \mathbb{K}_n^{\ell \times p}$ is obtained by first transposing and conjugating each one of the frontal slices of $\mathscr{A}$, and then reversing the order of the conjugated transposed frontal slices 2 through n. The tensor conjugate transpose has similar properties as the matrix transpose. For instance, if $\mathscr{A}$ and $\mathscr{B}$ are tensors such that $\mathscr{A} * \mathscr{B}$ and $\mathscr{B}^H * \mathscr{A}^H$ are defined, then $(\mathscr{A} * \mathscr{B})^H = \mathscr{B}^H * \mathscr{A}^H$.

(v) A third order tensor is said to be f-diagonal if its frontal slices in the Fourier domain are diagonal matrices.

(vi) A third order tensor is said to be f-upper-triangular if its frontal slices in the Fourier domain are upper triangular matrices.

(vii) The tensor $\mathscr{A} \in \mathbb{K}_n^{\ell \times \ell}$ is said to be f-Hermitian if $\mathscr{A}^H = \mathscr{A}$. In order for $\mathscr{A}$ to be f-Hermitian each frontal slice of $\widehat{\mathscr{A}}$ has to be a Hermitian matrix.

(viii) A tensor $\mathscr{Q} \in \mathbb{K}_n^{\ell\times\ell}$ is said to be f-orthogonal (or f-unitary) if $\mathscr{Q}^H * \mathscr{Q} = \mathscr{Q} * \mathscr{Q}^H = \mathscr{I}_\ell$.

(ix) A tensor $\mathscr{A} \in \mathbb{K}_n^{\ell\times\ell}$ is invertible if there is a tensor $\mathscr{A}^{-1} \in \mathbb{K}_n^{\ell\times\ell}$ such that $\mathscr{A}^{-1} * \mathscr{A} = \mathscr{A} * \mathscr{A}^{-1} = \mathscr{I}_\ell$. This definition implies that a third order tensor is invertible under the t-product if and only if each frontal slice in the Fourier domain is invertible.

(x) Let p be a positive integer and $\mathscr{A} \in \mathbb{K}_n^{\ell\times\ell}$. Then $\mathscr{A}^p = \mathscr{A} * \ldots * \mathscr{A}$. We define $\mathscr{A}^0 = \mathscr{I}_\ell$.

(xi) Let $\mathscr{A} \in \mathbb{K}_n^{\ell\times p}$. Then $\mathscr{A} \not\equiv \mathbf{0}$ means that $\widehat{\mathscr{A}^{(i)}} \neq 0$ for every $i = 1, 2, \ldots, n$.

In the following we will require the t-product of two tubes. This operation has been described in [26]. The definition suggests how the quotient of two tubes can be defined.

Definition 2. *Let $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{K}_n$. Then*

$$\boldsymbol{a} * \boldsymbol{b} = \texttt{ifft}(\widehat{\boldsymbol{a}} \circ \widehat{\boldsymbol{b}}, [], 3),$$

where $\circ$ denotes the entry-wise product of the vectors and, as usual, $\widehat{\boldsymbol{a}}$ and $\widehat{\boldsymbol{b}}$ stand for the Fourier transform of the tubes $\boldsymbol{a}$ and $\boldsymbol{b}$, respectively. Note that the tube product commutes.

Assume, in addition, that all entries of $\widehat{\boldsymbol{b}}$ are nonvanishing, i.e., $\boldsymbol{b} \not\equiv \mathbf{0}$; cf. Proposition 2(xi). Then we can define

$$\frac{\boldsymbol{a}}{\boldsymbol{b}} = \texttt{ifft}(\widehat{\boldsymbol{a}} \div \widehat{\boldsymbol{b}}, [], 3),$$

where $\div$ denotes entry-wise division.

It follows from Definition 2 that for $\vec{\mathscr{U}} \in \mathbb{K}_n^p$ and $\boldsymbol{b} \in \mathbb{K}_n \backslash \mathbf{0}$, we have

$$\frac{\vec{\mathscr{U}}}{\boldsymbol{b}} = \vec{\mathscr{U}} * \left(\frac{\boldsymbol{e}}{\boldsymbol{b}}\right).$$

It is convenient to introduce the following space.

Definition 3. *Let $\mathbb{G} = \{\vec{\mathscr{A}_1}, \vec{\mathscr{A}_2}, \ldots, \vec{\mathscr{A}_p}\} \subset \mathbb{K}_n^\ell$ be the set of lateral slices of the tensor $\mathscr{A} \in \mathbb{K}_n^{\ell\times p}$. Then the set of the t-linear combinations of these slices is a lateral slice space given by*

$$\begin{aligned} span_{\mathrm{t}}\,\{\mathbb{G}\} &= span_{\mathrm{t}}\left\{\vec{\mathscr{A}_1}, \vec{\mathscr{A}_2}, \ldots, \vec{\mathscr{A}_p}\right\} \\ &= \left\{\vec{\mathscr{C}} \in \mathbb{K}_n^\ell : \ \vec{\mathscr{C}} = \sum_{i=1}^{p} \vec{\mathscr{A}_i} * \boldsymbol{b}_i, \quad where \ \ \boldsymbol{b}_i \in \mathbb{K}_n, \ i = 1, 2, \ldots, p\right\}. \end{aligned}$$

3. Tensor factorizations

Many matrix factorizations can be extended to third-order tensors under the t-product. We will review two known factorizations and introduce a new one that will be used in the sequel.

Theorem 1. *(t-SVD) ([26]) Let $\mathscr{A} \in \mathbb{K}_n^{\ell\times p}$. Then, using the t-product, $\mathscr{A}$ can be factored into three third-order tensors,*

$$\mathscr{A} = \mathscr{U} * \mathscr{S} * \mathscr{V}^H, \tag{4}$$

where $\mathscr{U} \in \mathbb{K}_n^{\ell\times \ell}$ and $\mathscr{V} \in \mathbb{K}_n^{p\times p}$ are f-orthogonal tensors and $\mathscr{S} \in \mathbb{K}_n^{\ell\times p}$ is an f-diagonal tensor. We refer to the factorization (4) *as the* singular tube decomposition.

The tubes $\boldsymbol{s}_i = \mathscr{S}_{i,i,:}$ for $i = 1, 2, \ldots, \min\{\ell, p\}$ of the tensor $\mathscr{S}$ are referred to as singular tubes, whereas $\vec{\mathscr{U}}_j = \mathscr{U}_{:,j,:}$ for $j = 1, 2, \ldots, \ell$ and $\vec{\mathscr{V}}_k = \mathscr{V}_{:,k,:}$ for $k = 1, 2, \ldots, p$ are called left and right singular (lateral) slices, respectively. The norm of the singular tubes $\boldsymbol{s}_i$, given by $\sigma_i = \|\boldsymbol{s}_i\|_F$ for $i = 1, 2, \ldots, \min\{\ell, p\}$, are referred to as the singular values of the tensor $\mathscr{A}$; they are enumerated in decreasing order. The singular triplets $\{\boldsymbol{s}_i, \vec{\mathscr{U}}_i, \vec{\mathscr{V}}_i\}_{i=1}^{\min\{\ell,p\}}$ satisfy

$$\mathscr{A} * \vec{\mathscr{V}}_i = \vec{\mathscr{U}}_i * \boldsymbol{s}_i, \quad \mathscr{A}^H * \vec{\mathscr{U}}_i = \vec{\mathscr{V}}_i * \boldsymbol{s}_i, \quad 1 = 1, 2, \ldots, \min\{\ell, p\}.$$

We note that equation (4) can be written as

$$\mathscr{A} = \sum_{i=1}^{\min\{\ell,p\}} \vec{\mathscr{U}}_i * \boldsymbol{s}_i * \vec{\mathscr{V}}_i^H.$$

Kilmer et al. [26] also describe the t-QR factorization of a tensor $\mathscr{A} \in \mathbb{K}_n^{\ell\times p}$,

$$\mathscr{A} = \mathscr{Q} * \mathscr{R},$$

where $\mathscr{Q} \in \mathbb{K}_n^{\ell\times\ell}$ is an f-orthogonal tensor and $\mathscr{R} \in \mathbb{K}_n^{\ell\times p}$ is an f-upper-triangular tensor.

Theorem 2. *Let the third-order tensor $\mathscr{A} \in \mathbb{K}_n^{p\times p}$ be invertible. Then LU factorization with partial pivoting can be applied in the Fourier domain, resulting in the t-LU decomposition*

$$\mathscr{P} * \mathscr{A} = \mathscr{L} * \mathscr{U}, \tag{5}$$

where $\mathscr{P} \in \mathbb{K}_n^{p\times p}$ is the Fourier transform of permutation matrices, $\mathscr{L} \in \mathbb{K}_n^{p\times p}$ is an f-unit-lower triangular tensor, and $\mathscr{U} \in \mathbb{K}_n^{p\times p}$ is an f-upper-triangular tensor.

Proof. Let $\widehat{\mathscr{A}} = \texttt{fft}(\mathscr{A}, [\,], 3)$. Then, for $i = 1, \ldots, n$, we apply LU factorization with partial pivoting to the frontal slices $\widehat{\mathscr{A}^{(i)}}$ of $\widehat{\mathscr{A}}$. This gives the decompositions

$$\widehat{\mathscr{P}^{(i)}}\widehat{\mathscr{A}^{(i)}} = \widehat{\mathscr{L}^{(i)}}\widehat{\mathscr{U}^{(i)}}, \quad i = 1, 2, \ldots, n,$$

where $\widehat{\mathscr{P}^{(i)}}$ is a permutation matrix, $\widehat{\mathscr{L}^{(i)}}$ is a unit lower triangular matrix, and $\widehat{\mathscr{U}^{(i)}}$ is an upper triangular matrix. These decompositions exist because the tensor $\mathscr{A}$ is nonsingular. The matrices $\widehat{\mathscr{P}^{(i)}}$, $\widehat{\mathscr{L}^{(i)}}$, and $\widehat{\mathscr{U}^{(i)}}$ are frontal slices of the tensors $\widehat{\mathscr{P}}$, $\widehat{\mathscr{L}}$, and $\widehat{\mathscr{U}}$. The tensors in the decomposition (5) are determined by

$$\mathscr{P} = \texttt{ifft}\left(\widehat{\mathscr{P}}, [\,], 3\right), \quad \mathscr{L} = \texttt{ifft}\left(\widehat{\mathscr{L}}, [\,], 3\right), \quad \mathscr{U} = \texttt{ifft}\left(\widehat{\mathscr{U}}, [\,], 3\right).$$

□

The following factorization generalizes the Hessenberg reduction for matrices. It will be used below in the computation of tensor eigentubes and eigenslices.

Theorem 3. *The tensor $\mathscr{A} \in \mathbb{K}_n^{p\times p}$ is unitarily similar to an f-upper-Hessenberg tensor, i.e.,*

$$\mathscr{H} = \mathscr{W}^H * \mathscr{A} * \mathscr{W},$$

where $\mathscr{H} \in \mathbb{K}_n^{p\times p}$ is an f-upper-Hessenberg tensor (all frontal slices in the Fourier domain are upper Hessenberg matrices) and $\mathscr{W} \in \mathbb{K}_n^{p\times p}$ is an f-unitary tensor.

Proof. Let $\widehat{\mathscr{A}} = \texttt{fft}(\mathscr{A}, [\,], 3)$. Then, for $i = 1, \ldots, n$, the frontal slice $\widehat{\mathscr{A}^{(i)}}$ of $\widehat{\mathscr{A}}$ is unitarily similar to a Hessenberg matrix H_i. We have

$$H_i = W_i^H \widehat{\mathscr{A}^{(i)}} W_i, \quad i = 1, 2, \ldots, n,$$

where the $H_i \in \mathbb{C}^{p\times p}$ are upper Hessenberg matrices and the $W_i \in \mathbb{C}^{p\times p}$ are unitary matrices.

Define the tensors $\mathscr{H}'$ and $\mathscr{W}'$ as follows:

$$\mathscr{H}'^{(i)} = H_i \text{ and } \mathscr{W}'^{(i)} = W_i, \quad i = 1, 2, \ldots, n.$$

Then

$$\mathscr{H}'^{(i)} = \left(\mathscr{W}'^{(i)}\right)^H \widehat{\mathscr{A}^{(i)}} \mathscr{W}'^{(i)}, \quad i = 1, 2, \ldots, n.$$

Consequently, defining the tensors $\mathscr{H} = \texttt{ifft}\,(\mathscr{H}', [\,], 3)$ and $\mathscr{W} = \texttt{ifft}\,(\mathscr{W}', [\,], 3)$, we conclude that

$$\mathscr{H} = \mathscr{W}^H * \mathscr{A} * \mathscr{W},$$

where $\mathscr{H}$ is an f-Hessenberg tensor since each frontal slice is an upper Hessenberg matrix in the Fourier domain; obviously the tensor $\mathscr{W}$ is unitary, since its frontal slices in the Fourier domain are unitary. □

Algorithm 2 provides details of the transformation of a tensor to f-upper-Hessenberg form.

Algorithm 2 t-Hessenberg reduction (`t-hess`)

```
Input: $\mathscr{A} \in \mathbb{K}_n^{p\times p}$.
Output: Hessenberg tensor $\mathscr{H} \in \mathbb{K}_n^{p\times p}$ and unitary tensor $\mathscr{W} \in \mathbb{K}_n^{p\times p}$.

1: $\widehat{\mathscr{A}} = \texttt{fft}\,(\mathscr{A}, [\,], 3)$.
2: for $i = 1$ to $n$ do
3:     $[\widehat{\mathscr{W}}^{(i)}, \widehat{\mathscr{H}}^{(i)}] = \texttt{hess}(\mathscr{A}^{(i)})$
4: end for
5: $\mathscr{W} = \texttt{ifft}\left(\widehat{\mathscr{W}}, [\,], 3\right)$, $\mathscr{H} = \texttt{ifft}\left(\widehat{\mathscr{H}}, [\,], 3\right)$
```

The MATLAB function `hess` used in line 3 of Algorithm 2 applies a unitary similarity transformation $\widehat{\mathscr{W}}^{(i)}$ to the matrix $\widehat{\mathscr{A}}^{(i)}$ to determine an upper Hessenberg matrix $\widehat{\mathscr{H}}^{(i)}$. In the case of real tensors we can use the symmetry property of the DFT.

Following Kilmer et al. [26], we introduce a bilinear form associated with the t-product. For lateral slices $\vec{\mathscr{X}}, \vec{\mathscr{Y}} \in \mathbb{K}_n^{\ell}$, we define

$$\langle \vec{\mathscr{X}}, \vec{\mathscr{Y}} \rangle = \vec{\mathscr{X}}^{H} * \vec{\mathscr{Y}} \in \mathbb{K}_n.$$

The value of this bilinear form is a tube and the quantity $\langle \vec{\mathscr{X}}, \vec{\mathscr{X}} \rangle$ is not necessarily real and positive. The next lemma gives some properties of this bilinear form.

Lemma 1. *([26]) Let $\vec{\mathscr{X}}, \vec{\mathscr{Y}}, \vec{\mathscr{Z}} \in \mathbb{K}_n^{p}$ and $\boldsymbol{a} \in \mathbb{K}_n$. Then the following properties hold:*

1. $\langle \vec{\mathscr{X}}, \vec{\mathscr{Y}} + \vec{\mathscr{Z}} \rangle = \langle \vec{\mathscr{X}}, \vec{\mathscr{Y}} \rangle + \langle \vec{\mathscr{X}}, \vec{\mathscr{Z}} \rangle$,
2. $\langle \vec{\mathscr{X}}, \vec{\mathscr{Y}} * \boldsymbol{a} \rangle = \boldsymbol{a} * \langle \vec{\mathscr{X}}, \vec{\mathscr{Y}} \rangle$,
3. $\langle \boldsymbol{a} * \vec{\mathscr{X}}, \vec{\mathscr{Y}} \rangle = \boldsymbol{a}^{H} * \langle \vec{\mathscr{X}}, \vec{\mathscr{Y}} \rangle$,
4. $\langle \vec{\mathscr{X}} . \vec{\mathscr{Y}} \rangle = \langle \vec{\mathscr{Y}}, \vec{\mathscr{X}} \rangle^{H}$.

A sequence $\left\{ \vec{\mathscr{X}}_1, \vec{\mathscr{X}}_2, \ldots, \vec{\mathscr{X}}_p \right\} \subset \mathbb{K}_n^{\ell}$ of $p \geqslant 2$ lateral slices is said to be f-orthogonal if

$$\langle \vec{\mathscr{X}}_i, \vec{\mathscr{X}}_j \rangle = \vec{\mathscr{X}}_i^{H} * \vec{\mathscr{X}}_j = \begin{cases} \alpha_i \mathbf{e} & \text{if } i = j, \;\; i, j = 1, 2, \ldots, p, \\ \mathbf{0} & \text{else,} \end{cases}$$

where $\boldsymbol{e}$ is the unit tube and $\alpha_i \in \mathbb{C}$. If $\alpha_i = 1$ for all $1 \leqslant i \leqslant p$, then the sequence is said to be f-orthonormal.

Let the lateral slice $\vec{\mathscr{Y}} \in \mathbb{K}_n^\ell$ be nonvanishing. We then can normalize $\vec{\mathscr{Y}}$ in order to obtain a slice $\vec{\mathscr{X}} \in \mathbb{K}_n^\ell$ with norm $\|\vec{\mathscr{X}}\| = 1$, i.e., $\vec{\mathscr{Y}} = \vec{\mathscr{X}} * \boldsymbol{a}$ with $\|\vec{\mathscr{Y}}\| = \|\boldsymbol{a}\|$, where

$$\|\vec{\mathscr{Y}}\| = \frac{\|\langle \vec{\mathscr{Y}}, \vec{\mathscr{Y}} \rangle\|_F}{\|\vec{\mathscr{Y}}\|_F}.$$

4. Tensor eigenpairs based on the t-product

This section discusses properties of tensor eigenpairs for third-order tensors defined with the aid of the t-product. Tensor eigenpairs have previously been considered by Kilmer et al. [26] and, in more detail, in the unpublished report [25]. This report also describes some of the properties shown below. We include discussions of properties already discussed in [25] for completeness.

Definition 4. *Let $\mathscr{A} \in \mathbb{K}_n^{p\times p}$. The tube $\boldsymbol{\lambda} \in \mathbb{K}_n$ is said to be an eigentube of $\mathscr{A}$ associated to a lateral slice $\vec{\mathscr{U}} \not\equiv \mathbf{0}$ in $\mathbb{K}_n^p$, if it satisfies*

$$\mathscr{A} * \vec{\mathscr{U}} = \vec{\mathscr{U}} * \boldsymbol{\lambda}. \tag{6}$$

The lateral slice $\vec{\mathscr{U}}$ is said to be an eigenslice *or* right eigenslice *of $\mathscr{A}$ associated with $\boldsymbol{\lambda}$. Moreover, the set $\{\boldsymbol{\lambda}, \vec{\mathscr{U}}\}$ is referred to as an eigenpair of $\mathscr{A}$.*

We remark that when $n = 1$, the third-order tensor reduces to a square matrix, and the eigentubes and eigenslices become eigenvalues and eigenvectors, respectively. The set of all eigentubes of $\mathscr{A}$ is referred to as the *spectrum* of $\mathscr{A}$ and is denoted by $\boldsymbol{\Lambda}(\mathscr{A})$. The *spectral radius* of $\mathscr{A}$ is defined as

$$\rho(\mathscr{A}) = \max_{\boldsymbol{\lambda}\in\boldsymbol{\Lambda}(\mathscr{A})} \|\boldsymbol{\lambda}\|_F\,.$$

Let $\boldsymbol{\lambda}$ be an eigentube of $\mathscr{A} \in \mathbb{K}_n^{p\times p}$. The slice $\vec{\mathscr{V}} \in \mathbb{K}_n^p$ such that

$$\mathscr{A}^H * \vec{\mathscr{V}} = \vec{\mathscr{V}} * \boldsymbol{\lambda}^H$$

is called a *left eigenslice* of $\mathscr{A}$ associated with $\boldsymbol{\lambda}$.

Proposition 3. *Let $\mathscr{A} \in \mathbb{K}_n^{p\times p}$ be a third-order tensor, and let $\boldsymbol{\lambda} \in \mathbb{K}_n$ and $\vec{\mathscr{U}} \in \mathbb{K}_n^p$. Then we have*

$$\mathscr{A} * \vec{\mathscr{U}} = \vec{\mathscr{U}} * \boldsymbol{\lambda} \Longleftrightarrow \widehat{\mathscr{A}}^{(i)}\widehat{\vec{\mathscr{U}}}^{(i)} = \widehat{\vec{\mathscr{U}}}^{(i)}\widehat{\boldsymbol{\lambda}}^{(i)},\ i = 1, 2, \ldots, n.$$

Example 1. *Let the tensor $\mathscr{A} \in \mathbb{R}^{2\times 2\times 2}$ be defined by its frontal slices*

$$\mathscr{A}^{(1)} = \begin{pmatrix} 1 & 0 \\ 2 & 1 \end{pmatrix}, \quad \mathscr{A}^{(2)} = \begin{pmatrix} 1 & 1 \\ 2 & 0 \end{pmatrix}.$$

The set of eigenvalues of $\widehat{\mathscr{A}}^{(1)}$ *is* $\{3.5616, -0.5616\}$*, and the set of eigenvalues of* $\widehat{\mathscr{A}}^{(2)}$ *is* $\{1, 0\}$*. By using Proposition 3, we obtain that* $\texttt{ifft}\left([3.5616, 1], [], 3\right)$*,* $\texttt{ifft}\left([-0.5616, 0], [], 3\right)$*,* $\texttt{ifft}\left([3.5616, 0], [], 3\right)$*, and* $\texttt{ifft}\left([-0.5616, 1], [], 3\right)$ *are eigentubes of* $\mathscr{A}$*. Here* $\texttt{ifft}\left([-0.5616, 0], [], 3\right)$ *is short-hand notation for* $\texttt{ifft}(a, [], 3)$*, where* $a(1,1,1) = -0.5616$ *and* $a(1,1,2) = 0$*. This determines a tube with frontal slices* -0.2808 *and* -0.2808*.*

The above example illustrates that a tensor may have numerous eigentubes, with many of them obtained by combining the eigenvalues of $\widehat{\mathscr{A}}^{(1)}$ and $\widehat{\mathscr{A}}^{(2)}$. We determine a subset of eigentubes by using the following ordering: Let $\mathscr{A} \in \mathbb{K}_n^{p\times p}$ be a third-order tensor, and let $\gamma_{1,i}, \gamma_{2,i}, \ldots, \gamma_{p,i}$ be the eigenvalues of $\widehat{\mathscr{A}}^{(i)}$ for $i = 1, 2, \ldots, n$, ordered so that

$$|\gamma_{j,i}| \geqslant |\gamma_{j+1,i}|, \quad j = 1, 2, \ldots, p-1.$$

We define the *ordered eigentubes* $\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \ldots, \boldsymbol{\lambda}_p$ of $\mathscr{A}$ as

$$\widehat{\boldsymbol{\lambda}}_j^{(i)} = \gamma_{j,i},\ i = 1, 2, \ldots, n,\ j = 1, 2, \ldots, p,$$

Notice that a tensor $\mathscr{A}_n^{p\times p}$ admits at most p ordered eigentubes. We are interested in these ordered eigentubes and will for simplicity in the remainder of this work refer to them as eigentubes.

We turn to the linear independence of lateral slices. Let $\{\vec{\mathscr{V}}_i\}_{i=1}^s \subset \mathbb{K}_n^p$ be a set of lateral slices. This set is t-linearly independent if and only if

$$\sum_{i=1}^{s} \vec{\mathscr{V}}_i * \boldsymbol{a}_i = \mathbf{0} \Longrightarrow \boldsymbol{a}_i = \mathbf{0}, \quad i = 1, 2, \ldots, s.$$

It is clear that the slices $\vec{\mathscr{V}}_1, \ldots, \vec{\mathscr{V}}_s$ in $\mathbb{K}_n^p$ are t-linearly independent if and only if the vectors $\widehat{\vec{\mathscr{V}}}_1^{(j)}, \ldots, \widehat{\vec{\mathscr{V}}}_s^{(j)}$ are linearly independent for $j = 1, 2, \ldots, n$. This leads to the following result.

Proposition 4. *A tensor* $\mathscr{A} \in \mathbb{K}_n^{p\times p}$ *is invertible under the t-product if it has* p *t-linearly independent lateral slices.*

We are in a position to define f-diagonalization of a third-order tensor.

Definition 5. *A third-order tensor* $\mathscr{A} \in \mathbb{K}_n^{p\times p}$ *is said to be f-diagonalizable under the t-product if and only if it is similar to an f-diagonal tensor, i.e., if and only if*

$$\mathscr{A} = \mathscr{X} * \mathscr{D} * \mathscr{X}^{-1}$$

for some invertible tensor $\mathscr{X} \in \mathbb{K}_n^{p\times p}$ *and an f-diagonal tensor* $\mathscr{D}$ *formed by the eigentubes.*

Proposition 5. *A tensor* $\mathscr{A} \in \mathbb{K}_n^{p\times p}$ *is f-diagonalizable if and only if it has* p *t-linearly independent eigenslices.*

Proposition 6. *Let $\vec{\mathscr{U}}$ be an eigenslice of $\mathscr{A}^H * \mathscr{A}$ associated with the eigentube $\boldsymbol{\lambda}$. Then $\mathscr{A} * \vec{\mathscr{U}}$ is an eigenslice of $\mathscr{A} * \mathscr{A}^H$ associated to $\boldsymbol{\lambda}$.*

Proof. We have $\mathscr{A}^H * \mathscr{A} * \vec{\mathscr{U}} = \vec{\mathscr{U}} * \boldsymbol{\lambda}$. Therefore, $\mathscr{A} * \mathscr{A}^H * \mathscr{A} * \vec{\mathscr{U}} = \mathscr{A} * \vec{\mathscr{U}} * \boldsymbol{\lambda}$. □

Theorem 4. *Let the tensor $\mathscr{A} \in \mathbb{K}_n^{p\times p}$ be f-Hermitian. Then all eigentubes $\boldsymbol{\lambda}$ of $\mathscr{A}$ are real.*

Proof. Let $\{\boldsymbol{\lambda}, \vec{\mathscr{U}}\}$ be an eigenpair of $\mathscr{A}$ with $\vec{\mathscr{U}} \not\equiv \mathbf{0}$. Then

$$\boldsymbol{\lambda} = \frac{\langle \vec{\mathscr{U}}, \mathscr{A} * \vec{\mathscr{U}} \rangle}{\langle \vec{\mathscr{U}}, \vec{\mathscr{U}} \rangle} = \frac{\langle \mathscr{A}^H * \vec{\mathscr{U}}, \vec{\mathscr{U}} \rangle}{\langle \vec{\mathscr{U}}, \vec{\mathscr{U}} \rangle} = \boldsymbol{\lambda}^H.$$

Since $\mathscr{A}$ is f-Hermitian, we have that each frontal slice of $\widehat{\mathscr{A}}$ is a Hermitian matrix. Thus the eigenvalues of each frontal slice of $\widehat{\mathscr{A}}$ are real. Each eigentube of $\mathscr{A}$ is given by $\boldsymbol{\lambda} = \texttt{ifft}(\widehat{\boldsymbol{\lambda}}, [\,], 3)$, where $\widehat{\boldsymbol{\lambda}}$ is a real conjugate-even tube. Consequently, $\boldsymbol{\lambda}$ is a real conjugate-even tube. □

Similarly as for matrices, we can relate eigentubes to tensor singular values. Let $\mathscr{A} \in \mathbb{K}_p^{n\times m}$ be a third-order tensor with tubal rank $r > 0$. Consider the analogue of the singular tube decomposition (4) of $\mathscr{A}$ with the singular tubes are ordered according to

$$\|\boldsymbol{s}_1\|_F \geqslant \|\boldsymbol{s}_2\|_F \geqslant \ldots \geqslant \|\boldsymbol{s}_r\|_F \geqslant \|\boldsymbol{s}_{r+1}\|_F = \ldots = \|\boldsymbol{s}_{\min\{m,n\}}\|_F = 0.$$

Then

$$\begin{array}{rcll} \mathscr{A} * \vec{\mathscr{V}}_i & = & \vec{\mathscr{U}}_i * \boldsymbol{s}_i,\ i = 1, 2, \ldots, r, & \mathscr{A} * \vec{\mathscr{V}}_i = 0,\ i = r+1, r+2, \ldots, m, \\ \mathscr{A}^H * \vec{\mathscr{U}}_i & = & \vec{\mathscr{V}}_i * \boldsymbol{s}_i,\ i = 1, 2, \ldots, r, & \mathscr{A}^H * \vec{\mathscr{U}}_i = 0,\ i = r+1, r+2, \ldots, n. \end{array}$$

This implies that $\vec{\mathscr{V}}_1, \ldots, \vec{\mathscr{V}}_m$ are eigenslices of $\mathscr{A}^H * \mathscr{A}$, $\vec{\mathscr{U}}_1, \ldots, \vec{\mathscr{U}}_m$ are eigenslices of $\mathscr{A} * \mathscr{A}^H$, and $\boldsymbol{s}_i^2$, $i = 1, 2, \ldots, r$, are nonzero eigentubes of $\mathscr{A}^H * \mathscr{A}$ and $\mathscr{A} * \mathscr{A}^H$.

Numerical methods for spectral factorization of a real nonsymmetric matrix use the real Schur form. We therefore are interested in the analogous factorization of third-order tensors.

Theorem 5 (Real t-Schur decomposition). *Let $\mathscr{A} \in \mathbb{R}^{p\times p\times n}$ be a real third-order tensor. There is a real f-orthogonal tensor $\mathscr{Q} \in \mathbb{R}^{p\times p\times n}$ such that*

$$\mathscr{Q} * \mathscr{A} * \mathscr{Q}^H = \mathscr{R} = \begin{pmatrix} \mathscr{R}_{1,1} & \mathscr{R}_{1,2} & \cdots & \cdots & \mathscr{R}_{1,m} \\ 0 & \mathscr{R}_{2,2} & \mathscr{R}_{2,2} & \cdots & \mathscr{R}_{2,m} \\ 0 & \cdots & \ddots & \cdots & \vdots \\ \vdots & \cdots & & & \vdots \\ 0 & \vdots & & & \mathscr{R}_{m,m} \end{pmatrix} \in \mathbb{R}^{p\times p\times n},$$

where the $\mathscr{R}_{i,i}$ are tensors of size $1 \times 1 \times n$ or $2 \times 2 \times n$. The tensor $\mathscr{R}$ therefore is said to be f-quasi-triangular.

Proof. The factorization follows from the real Schur factorization of the matrices in each frontal slice of the tensor $\widehat{\mathscr{A}}$, and from the fact that the first frontal slice of $\widehat{\mathscr{A}}$ is real, because $\widehat{\mathscr{A}^{(1)}} = \dfrac{1}{\sqrt{n}}\displaystyle\sum_{i=1}^{n}\mathscr{A}^{(i)}$. □

For $\mathscr{A} \in \mathbb{K}_n^{p\times p}$, let

$$r_i = \dim\left(\mathrm{Null}\left(\widehat{\mathscr{A}^{(i)}}\right)\right), \quad i = 1, 2, \ldots, n, \quad r = \min_{1 \leqslant i \leqslant n} r_i.$$

Definition 6. *Let $\mathscr{A} \in \mathbb{K}_n^{\ell\times p}$. The range of $\mathscr{A}$ is the t-span of the p lateral slices of $\mathscr{A}$, i.e.,*

$$Range\,(\mathscr{A}) = \left\{\mathscr{A} * \vec{\mathscr{X}} \,:\, \vec{\mathscr{X}} \in \mathbb{K}_n^p\right\},$$

and the null space of $\mathscr{A}$ is defined as

$$Null_{\mathrm{t}}\,(\mathscr{A}) = \left\{ \vec{\mathscr{X}}_1, \ldots, \vec{\mathscr{X}}_r \in \mathbb{K}_n^p \,:\, \widehat{\vec{\mathscr{X}}}_j^{(i)} \in Null\left(\widehat{\mathscr{A}^{(i)}}\right) \text{ for } i = 1, 2, \ldots, n, \right.$$
$$\left. \text{with } \|\widehat{\vec{\mathscr{X}}}_j^{(i)}\|_F \geqslant \|\widehat{\vec{\mathscr{X}}}_{j+1}^{(i)}\|_F,\ j = 1, 2, \ldots, r_i \right\}.$$

It is clear from Definition 6 that if $\vec{\mathscr{X}} \in \mathbb{K}_n^p$ is in the null space of a tensor $\mathscr{A} \in \mathbb{K}_n^{p\times p}$, then $\mathscr{A} * \vec{\mathscr{X}} = 0$.

Let $\mathscr{A} \in \mathbb{K}_n^{p\times p}$ and let $\boldsymbol{\lambda}_i$ be the i^{th} eigentube of $\mathscr{A}$. Define $\boldsymbol{\Lambda}_i = \text{t-diag}\left[\boldsymbol{\lambda}_i, \boldsymbol{\lambda}_i, \ldots, \boldsymbol{\lambda}_i\right]$.

Definition 7. *Assume that the tensor $\mathscr{A} \in \mathbb{K}_n^{p\times p}$ has ℓ distinct eigentubes $\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \ldots, \boldsymbol{\lambda}_\ell$. The index of the eigentube $\boldsymbol{\lambda}_i$ is the smallest integer such that*

$$Null_{\mathrm{t}}\,(\mathscr{A} - \boldsymbol{\Lambda}_i)^{\ell_i} = Null_{\mathrm{t}}\,(\mathscr{A} - \boldsymbol{\Lambda}_i)^{k}, \ \ \forall k \geqslant \ell_i.$$

Let ℓ_i be the index of the eigentube $\boldsymbol{\lambda}_i$ of $\mathscr{A}$. Then ℓ_i is the maximum index of the eigenvalues $\widehat{\boldsymbol{\lambda}}_i^{(j)}$ of the matrix $\widehat{\mathscr{A}^{(j)}}$ for $j = 1, 2, \ldots, n$.

Proposition 7. *Let $\vec{\mathscr{X}} \in \mathbb{K}_n^p$. Then $\vec{\mathscr{X}}$ can be represented as*

$$\vec{\mathscr{X}} = \sum_{i=1}^{\ell} \vec{\mathscr{X}_i},$$

where $\vec{\mathscr{X}_i} \in Null_{\mathrm{t}}\,(\mathscr{A} - \boldsymbol{\Lambda}_i)^{\ell_i}$ and ℓ_i is the index of $\boldsymbol{\lambda}_i$ for $i = 1, 2, \ldots, \ell$.

Proof. Let $\vec{\mathscr{X}} \in \mathbb{K}_n^p$ and let $\widehat{\vec{\mathscr{X}}}$ denote the associated tensor in the Fourier domain. The frontal slices $\widehat{\vec{\mathscr{X}}}^{(k)}$, $k = 1, 2, \ldots, n$, of $\widehat{\vec{\mathscr{X}}}$ live in $\mathbb{C}^p$. There is a unique

$$x_{j,k} \in \mathrm{Null}\left(\widehat{\mathscr{A}^{(k)}} - \widehat{\boldsymbol{\lambda}}_j^{(k)} I\right)^{\ell_i},$$

such that

$$\widehat{\vec{\mathscr{X}}}^{(k)} = \sum_{j=1}^{\ell} x_{j,k}.$$

The sequences $\{x_{1,1}, \ldots, x_{\ell,1}\}$, ..., $\{x_{1,n}, \ldots, x_{\ell,n}\}$ determine the sequence of lateral slices $\{\vec{\mathscr{X}}_1, \ldots, \vec{\mathscr{X}}_\ell\}$, such that

$$\vec{\mathscr{X}} = \sum_{j=1}^{\ell} \vec{\mathscr{X}}_j,$$

with $\vec{\mathscr{X}}_j \in \mathrm{Null_t}\,(\mathscr{A} - \mathbf{\Lambda}_i)^{\ell_i}$. □

5. Methods for eigenpair computation of third-order tensors

This section describes several methods for computing one or several eigenpairs of third-order tensors.

5.1. Single lateral slice iteration

The power method is one of the oldest and simplest techniques for computing the eigenvalue of largest magnitude (if there is such a unique eigenvalue) and an associated eigenvector of a square matrix; see, e.g., [22]. It is also known as von Mises iteration [36]. Presently, this method finds applications to pagerank computations; see [5, 10]. We describe a t-power method for computing the eigentube of largest magnitude of a third-order tensor, and discuss some of its properties. These properties are well known in matrix computations. Kilmer et al. [26] describe a power method with a different normalization of the eigentubes; see below.

5.2. The t-power method

This method determines a sequence of lateral slices $\mathscr{A}^k * \vec{\mathscr{V}}_0$, $k = 1, 2, \ldots$, suitably scaled, where $\vec{\mathscr{V}}_0 \in \mathbb{K}_n^p$. Under suitable conditions on $\mathscr{A}$ and $\vec{\mathscr{V}}_0$ (see below), the sequence generated converges to an eigenslice associated with the eigentube of largest norm. Here and below "norm" refers to the Frobenius norm. The main advantage of the t-power method, when compared with other methods for computing eigenpairs, is the fairly small amount of computer memory required. Algorithm 3 describes the computations with the t-power method.

Algorithm 3 The t-power method

Input: $\mathscr{A} \in \mathbb{K}_n^{p\times p}$, $\vec{\mathscr{V}}_0 \in \mathbb{K}_n^p$ nonzero.
Output: An eigenpair $\left\{\boldsymbol{\lambda}, \vec{\mathscr{U}}\right\}$ of $\mathscr{A}$, where $\boldsymbol{\lambda}$ is the eigentube of largest norm.

```
1: for $k = 1, 2, \dots$ until convergence do
2:     $\boldsymbol{\alpha}_k = \text{t-}\max(\mathscr{A} * \vec{\mathscr{V}}_{k-1})$
3:     $\vec{\mathscr{V}}_k = (\mathscr{A} * \vec{\mathscr{V}}_{k-1})/\boldsymbol{\alpha}_k$
4: end for
5: $\vec{\mathscr{U}} = \vec{\mathscr{V}}_k$, $\boldsymbol{\lambda} = \boldsymbol{\alpha}_k$
```

The stopping criterion for Algorithm 3 will be discussed in Section 6. The function t- max of a lateral slice in line 2 of Algorithm 3 returns the tube of largest norm of all tubes in the lateral slice. Algorithm 4 provides the details of this computation.

Algorithm 4 t- max

non-vanishin tensor **Input:** $\vec{\mathscr{V}} \in \mathbb{K}_n^p$.
Output: The tube of largest norm of $\vec{\mathscr{V}}$.

```
$\widehat{\mathscr{V}} = \texttt{fft}\left(\vec{\mathscr{V}}, [], 3\right)$.
for $i = 1, 2, \dots, n$ do
    $[k, \sim] = \max(\left|\widehat{\mathscr{V}}^{(i)}\right|)$
    $\widehat{\boldsymbol{a}}^{(i)} = \widehat{\mathscr{V}}(k, :, i)$
end for
$\boldsymbol{a} = \texttt{ifft}\,(\widehat{\boldsymbol{a}}, [], 3)$.
```

We remark that the lateral slices $\vec{\mathscr{V}}_k$ in the power method described in [26] are normalized so that $\|\vec{\mathscr{V}}_k\| = 1$. We will illustrate the difference between this normalization and the one used in Algorithm 3 in Section 6. Moreover, the power method presented in [26] evaluates $\vec{\mathscr{V}}_k^H * (\mathscr{A} * \vec{\mathscr{V}}_k)$ every iteration; this is not necessary.

The following theorems generalizes the classical convergence result for the matrix power method to the t-power method. The proof is analogous to the matrix case and therefore is omitted.

Theorem 6. *Let $\mathscr{A} \in \mathbb{K}_n^{p\times p}$ have only one eigentube, $\boldsymbol{\lambda}_1$, of largest norm. The initial lateral slice $\vec{\mathscr{V}}_0$ for the t-power method either has no tube in the invariant subspace associated with $\boldsymbol{\lambda}_1$, or the sequence $\vec{\mathscr{V}}_k$, $k = 1, 2, \dots$, generated by Algorithm 3 converges to an eigenslice associated with $\boldsymbol{\lambda}_1$, and the sequence $\boldsymbol{\alpha}_k$, $k = 1, 2, \dots$, converges to $\boldsymbol{\lambda}_1$.*

Theorem 7. *Let $\mathscr{A} \in \mathbb{K}_n^{p\times p}$ have p distinct eigentubes $\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \dots, \boldsymbol{\lambda}_p$ ordered decreasingly, i.e., $\|\boldsymbol{\lambda}_1\|_F > \|\boldsymbol{\lambda}_2\|_F \geqslant \dots \geqslant \|\boldsymbol{\lambda}_p\|_F$. Let the lateral slice $\vec{\mathscr{V}}_0$ have* *a component in the invariant subspace associated with $\boldsymbol{\lambda}_1$. Then the sequence $\vec{\mathscr{V}_k}$, $k = 1, 2, \dots$, generated by Algorithm 3 converges to an eigenslice associated with the eigentube $\boldsymbol{\lambda}_1$.*

Proof. By Proposition 7, we have

$$\vec{\mathscr{V}_0} = \sum_{i=1}^{p} \vec{\mathscr{X}_i} * \boldsymbol{c}_i,$$

where $\vec{\mathscr{X}_i} \in \mathrm{Null_t}\left(\mathscr{A} - \boldsymbol{\Lambda}_i\right)$. Since $\vec{\mathscr{V}_0}$ has a component in the invariant subspace associated with $\boldsymbol{\lambda}_1$, it follows that $\boldsymbol{c}_1 \neq 0$.

We are interested in the space spanned by $\vec{\mathscr{V}_k}$. This space is independent of the scaling factors $\boldsymbol{\alpha}_k$ in Algorithm 3, which are assumed to be nonsingular, i.e., $1/\boldsymbol{\alpha}_k$ exists. Therefore, we may in this proof ignore the factors $\boldsymbol{\alpha}_k$ for all k. We then have for $k = 1, 2, \dots$ that

$$\vec{\mathscr{V}_k} = \mathscr{A}^k * \vec{\mathscr{V}_0} = \sum_{i=1}^{p} \vec{\mathscr{X}_i} * \boldsymbol{c}_i * \boldsymbol{\lambda}_i^k = \vec{\mathscr{X}_1} * \boldsymbol{c}_1 * \boldsymbol{\lambda}_1^k + \sum_{i=2}^{p} \vec{\mathscr{X}_i} * \boldsymbol{c}_i * \left(\frac{\boldsymbol{\lambda}_i}{\boldsymbol{\lambda}_1}\right)^k.$$

Therefore,

$$\begin{aligned} \left\| \vec{\mathscr{V}_k} - \vec{\mathscr{X}_1} * \boldsymbol{c}_1 * \boldsymbol{\lambda}_1^k \right\|_F &= \left\| \sum_{i=2}^{p} \vec{\mathscr{X}_i} * \boldsymbol{c}_i * \left(\frac{\boldsymbol{\lambda}_i}{\boldsymbol{\lambda}_1}\right)^k \right\|_F \\ &\leqslant \sum_{i=2}^{p} \left\| \vec{\mathscr{X}_i} * \boldsymbol{c}_i * \left(\frac{\boldsymbol{\lambda}_i}{\boldsymbol{\lambda}_1}\right)^k \right\|_F \\ &\leqslant \sum_{i=2}^{p} \left\| \frac{\boldsymbol{\lambda}_i}{\boldsymbol{\lambda}_1} \right\|_F^k \left\| \vec{\mathscr{V}_i} * \boldsymbol{c}_i \right\|_F. \end{aligned}$$

Since $\|\boldsymbol{\lambda}_i\|_F < \|\boldsymbol{\lambda}_1\|_F$ for all $i > 1$, the sum in the right-hand side converges to zero as k increases. It follows that

$$\lim_{k \to +\infty} \vec{\mathscr{V}_k} = \lim_{k \to +\infty} \vec{\mathscr{X}_1} * \boldsymbol{c}_1 * \boldsymbol{\lambda}_1^k.$$

This shows the theorem. □

We can see from the proof of Theorem 7, that the rate of convergence is larger, the smaller the quotients $\|\boldsymbol{\lambda}_i\|_F / \|\boldsymbol{\lambda}_1\|_F$ are for $i = 2, 3, \dots, p$. The rate of convergence of the t-power method may be improved by applying the method to a shifted tensor $\mathscr{A} + \boldsymbol{\Sigma}$, where

$$\boldsymbol{\Sigma} = [\boldsymbol{\sigma}, \boldsymbol{\sigma}, \dots, \boldsymbol{\sigma}] \in \mathbb{K}_n^{p \times p},$$

for a suitable shift $\boldsymbol{\sigma}$, instead of applying the method to $\mathscr{A}$. This follows from the proof of Theorem 7. We omit the details.

5.3. The inverse t-power method

In its simplest form this method can be applied to compute an eigenslice associated with the eigentube of smallest norm (if there is a unique eigentube with this property). Then the method is obtained by applying the t-power method to the tensor $\mathscr{A}^{-1}$. T his yields the sequence

$$\vec{\mathscr{V}}_k = (\mathscr{A}^{-1} * \vec{\mathscr{V}}_{k-1})/\boldsymbol{\alpha}_k, \quad k = 1, 2, \dots ,$$

of lateral slices, where $\vec{\mathscr{V}}_0$ is an initial user-supplied slice. The sequence converges to the lateral slice associated with the eigentube of largest norm of $\mathscr{A}^{-1}$, if this eigentube is unique and if $\vec{\mathscr{V}}_0$ contains a component of this slice. Since the eigentubes of $\mathscr{A}$ are the inverse of the eigentubes of $\mathscr{A}^{-1}$, we obtain convergence to the lateral slice associated with the eigentube of smallest norm of $\mathscr{A}$. Algorithm 5 describes a shifted version that can be used to determine an eigenslice associated with the eigentube that is closest to $\boldsymbol{\sigma}$, if this eigentube is unique and the initial slice $\vec{\mathscr{V}}_0$ contains a component of this eigenslice.

Algorithm 5 The shifted inverse t-power method

Input: $\mathscr{A} \in \mathbb{K}_n^{p\times p}$, $\vec{\mathscr{V}}_0 \in \mathbb{K}_n^{p}$, and $\boldsymbol{\sigma} \in \mathbb{K}_n$. Define $(\boldsymbol{\Sigma} = [\boldsymbol{\sigma}, \boldsymbol{\sigma}, \dots, \boldsymbol{\sigma}] \in \mathbb{K}_n^{p\times p})$.

Output: The eigenpair $\{\boldsymbol{\lambda}, \vec{\mathscr{U}}\}$ of $\mathscr{A}$, where $\boldsymbol{\lambda}$ is the eigentube closest to $\boldsymbol{\sigma}$.

```
1: for $k = 1, 2, \dots$ until convergence do
2:     Evaluate $\vec{\mathscr{W}} = (\mathscr{A} - \boldsymbol{\Sigma})^{-1} * \vec{\mathscr{V}}_{k-1}$
3:     $\boldsymbol{\alpha}_k = \text{t-}\max(\vec{\mathscr{W}})$
4:     $\vec{\mathscr{V}}_k = \vec{\mathscr{W}}/\boldsymbol{\alpha}_k$
5: end for
6: $\vec{\mathscr{U}} = \vec{\mathscr{V}}_k$, $\boldsymbol{\lambda} = \dfrac{\boldsymbol{e}}{\boldsymbol{\alpha}_k + \boldsymbol{\sigma}}$
```

The evaluation of $\vec{\mathscr{W}}$ in line 2 of Algorithm 5 can be carried out, e.g., by using a t-QR, t-LU, or t-Choleski factorization of $\mathscr{A} - \boldsymbol{\Sigma}$; the t-Choleski factorization is described in [33].

5.4. Deflation

Deflation is a well known technique in matrix computation. It makes it possible to use the power method to compute more eigenpairs of a matrix than the pair associated with the eigenvalue of largest magnitude. This section describes how deflation can be used together with the t-power method.

Let the tensor $\mathscr{A} \in \mathbb{K}_n^{p\times p}$ have a unique eigentube $\boldsymbol{\lambda}_1$ of largest norm, and let $\vec{\mathscr{U}}_1$ be an associated eigenslice. Assume that the eigenpair $\{\boldsymbol{\lambda}_1, \vec{\mathscr{U}}_1\}$ has been computed, e.g., by the t-tensor power method. The idea behind deflation is to apply Algorithm 3 to the tensor

$$\mathscr{A}_1 = \mathscr{A} - \vec{\mathscr{U}}_1 * \boldsymbol{\sigma} * \vec{\mathscr{V}}^H, \tag{7}$$

where $\vec{\mathscr{V}} \in \mathbb{K}_n^{p}$ is an arbitrary lateral slice such that $\vec{\mathscr{V}}^H * \vec{\mathscr{U}}_1 = \boldsymbol{e}$, and $\boldsymbol{\sigma} \in \mathbb{K}_n$ is an appropriate shift. The tensor $\mathscr{A}_1$ has the same eigentubes as $\mathscr{A}$ except for $\boldsymbol{\lambda}_1$, which is transformed to $\boldsymbol{\lambda}_1 - \boldsymbol{\sigma}$, as is shown in the next theorem.

Theorem 8. *Let* $\mathscr{A} \in \mathbb{K}_n^{p\times p}$ *have the eigenpairs* $\{\boldsymbol{\lambda}_i, \vec{\mathscr{U}_i}\}_{i=1}^{\ell}$*, and let the tube* $\boldsymbol{\sigma} \in \mathbb{K}_n$ *be such that* $\boldsymbol{\lambda}_i \not\equiv \boldsymbol{\lambda}_1 - \boldsymbol{\sigma}$*,* $i = 1, 2, \ldots, \ell$*. Then the tensor* $\mathscr{A}_1$*, defined by* (7)*, has the eigenpairs*

$$\left\{ \left(\boldsymbol{\lambda}_1 - \boldsymbol{\sigma}, \vec{\mathscr{U}_1}\right), \left(\boldsymbol{\lambda}_i, \vec{\mathscr{U}_i} + \vec{\mathscr{U}_1} * \gamma_i\right)_{i=2}^{\ell} \right\},$$

with

$$\gamma_i = \frac{\boldsymbol{\sigma} * \vec{\mathscr{V}}^H * \vec{\mathscr{U}_i}}{\boldsymbol{\sigma} - (\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_i)}, \quad i = 2, 3, \ldots, \ell.$$

Proof. First note that $\boldsymbol{\lambda}_i \not\equiv \boldsymbol{\lambda}_1 - \boldsymbol{\sigma}$ means that

$$\widehat{\boldsymbol{\lambda}}_i^{(j)} \neq \widehat{\boldsymbol{\lambda}}_1^{(j)} - \widehat{\boldsymbol{\sigma}}^{(j)}, \quad 1 \leqslant j \leqslant n.$$

Let $\boldsymbol{\sigma}$ be such that $\boldsymbol{\lambda}_i \not\equiv \boldsymbol{\lambda}_1 - \boldsymbol{\sigma}$. We will show that $\{\boldsymbol{\lambda}_1 - \boldsymbol{\sigma}, \vec{\mathscr{U}_1}\}$ is an eigenpair of $\mathscr{A}_1$ and that the deflation procedure preserves the eigentubes $\boldsymbol{\lambda}_2, \boldsymbol{\lambda}_3, \ldots, \boldsymbol{\lambda}_\ell$. We have

$$\begin{aligned} \mathscr{A}_1 * \vec{\mathscr{U}_1} = \left(\mathscr{A} - \vec{\mathscr{U}_1} * \boldsymbol{\sigma} * \vec{\mathscr{V}}^H\right) * \vec{\mathscr{U}_1} &= \mathscr{A} * \vec{\mathscr{U}_1} - \vec{\mathscr{U}_1} * \boldsymbol{\sigma} * \vec{\mathscr{V}}^H * \vec{\mathscr{U}_1} \\ &= \vec{\mathscr{U}_1} * \boldsymbol{\lambda}_1 - \vec{\mathscr{U}_1} * \boldsymbol{\sigma} \\ &= \vec{\mathscr{U}_1} * (\boldsymbol{\lambda}_1 - \boldsymbol{\sigma}). \end{aligned}$$

On the other hand, let $\vec{\mathscr{W}_j}$ be the left eigenslice associated with $\boldsymbol{\lambda}_j$ for $j = 2, 3, \ldots, \ell$. Then we have

$$\mathscr{A}^H * \vec{\mathscr{W}_j} = \vec{\mathscr{W}_j} * \boldsymbol{\lambda_j},$$

and since $\{\vec{\mathscr{W}_j}\}_{j=2}^{\ell}$ are f-orthogonal to $\vec{\mathscr{U}_1}$, we get

$$\begin{aligned} \mathscr{A}_1^H * \vec{\mathscr{W}_j} = \left(\mathscr{A} - \vec{\mathscr{U}_1} * \boldsymbol{\sigma} * \vec{\mathscr{V}}^H\right)^H * \vec{\mathscr{W}_j} &= \left(\mathscr{A}^H - \vec{\mathscr{V}} * \boldsymbol{\sigma}^H * \vec{\mathscr{U}_1}^H\right) * \vec{\mathscr{W}_j} \\ &= \vec{\mathscr{W}_j} * \boldsymbol{\lambda}_j, \quad j = 2, 3, \ldots, \ell. \end{aligned}$$

Thus, deflation preserves the left eigenslices $\vec{\mathscr{W}_2}, \vec{\mathscr{W}_3}, \ldots, \vec{\mathscr{W}_\ell}$ and the eigentubes $\boldsymbol{\lambda}_2, \boldsymbol{\lambda}_3, \ldots, \boldsymbol{\lambda}_\ell$.

Since $\boldsymbol{\lambda}_2, \boldsymbol{\lambda}_3, \ldots, \boldsymbol{\lambda}_\ell$ are eigentubes of $\mathscr{A}_1$, we will now construct eigenslices associated with these eigentubes. Let for $i = 2, 3, \ldots, \ell$,

$$\breve{\vec{\mathscr{U}_i}} = \vec{\mathscr{U}_i} - \vec{\mathscr{U}_1} * \boldsymbol{\gamma}_i.$$

It follows from $\mathscr{A}_1 * \breve{\vec{\mathscr{U}_i}} = \check{\vec{\mathscr{U}}}_i * \boldsymbol{\lambda}_i$ that $\boldsymbol{\gamma}_i$ is given by

$$\gamma_i = \frac{\boldsymbol{\sigma} * \vec{\mathscr{V}}^H * \vec{\mathscr{U}_i}}{\boldsymbol{\sigma} - (\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_i)}, \quad i = 2, 3, \ldots, \ell,$$

and $\boldsymbol{\gamma}_1 = 0$ such that $\boldsymbol{\sigma} \not\equiv \boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_i$. □

Deflation techniques for matrices have been discussed in several works, see, e.g., Saad [35, Chapter 4.2], and the choice of the analogue of the slice $\vec{\mathscr{V}}$ has received some attention. Analyses for the matrix case suggest that we let $\vec{\mathscr{V}} = \vec{\mathscr{W}}_1$, where $\vec{\mathscr{W}}_1$ is the left eigenslice of $\mathscr{A}$, or $\vec{\mathscr{V}} = \vec{\mathscr{U}}_1$. The latter choice preserves the t-Schur lateral slices.

Proposition 8. *Let $\vec{\mathscr{U}}_1$ be an eigenslice of $\mathscr{A}$ of norm 1 associated with $\boldsymbol{\lambda}_1$ and let $\mathscr{A}_1 = \mathscr{A} - \vec{\mathscr{U}}_1 * \boldsymbol{\sigma} * \vec{\mathscr{U}}_1^H$. Then the eigentubes of $\mathscr{A}_1$ are $\widetilde{\boldsymbol{\lambda}}_1 = \boldsymbol{\lambda}_1 - \boldsymbol{\sigma}$ and $\widetilde{\boldsymbol{\lambda}}_j = \boldsymbol{\lambda}_j$, $j = 2, 3, \ldots, \ell$. Moreover, the t-Schur lateral slices associated with $\widetilde{\boldsymbol{\lambda}}_j$, $j = 1, 2, \ldots, \ell$ are identical with those of $\mathscr{A}$.*

Proof. Let $\mathscr{A} * \mathscr{U} = \mathscr{U} * \mathscr{R}$ be the t-Schur factorization of the tensor $\mathscr{A}$, where $\mathscr{U} \in \mathbb{K}_n^{p\times p}$ is an f-orthogonal tensor and $\mathscr{R} \in \mathbb{K}_n^{p\times p}$ is an upper f-triangular tensor. Thus,

$$\begin{aligned} \mathscr{A}_1 * \mathscr{U} &= \left(\mathscr{A} - \vec{\mathscr{U}}_1 * \boldsymbol{\sigma} * \vec{\mathscr{U}}_1^H\right) * \mathscr{U} \\ &= \mathscr{U} * \mathscr{R} - \vec{\mathscr{U}}_1 * \boldsymbol{\sigma} * \vec{\mathscr{E}}_1^H \\ &= \mathscr{U} * \left(\mathscr{R} - \vec{\mathscr{E}}_1 * \boldsymbol{\sigma} * \vec{\mathscr{E}}_1^H\right), \end{aligned}$$

which is the desired result since $\mathscr{R} - \vec{\mathscr{E}}_1 * \boldsymbol{\sigma} * \vec{\mathscr{E}}_1^H$ is an upper f-triangular tensor. □

Deflation can be applied multiple times using several vectors. For the matrix case this is described in, e.g., [35, Chapter 4.2.3]. The techniques can be adapted to the tensor case. We omit the details.

5.5. The tensor subspace iteration method

Subspace iteration is a powerful technique for computing a few eigenpairs with the largest eigenvalues of a large matrix; see, e.g., [28, 35]. This section describes the application of subspace iteration to the computation of a few eigenpairs with the largest eigentubes of a large tensor $\mathscr{A} \in \mathbb{K}_n^{p\times p}$. The method is described by Algorithm 6.

Algorithm 6 t-subspace iteration

Input: $\mathscr{A} \in \mathbb{K}_n^{p\times p}$, $\mathscr{X}_0 \in \mathbb{K}_n^{p\times s}$, integer $q > 0$.
Output: Upper f-triangular tensor $\mathscr{R} \in \mathbb{K}_n^{s\times s}$ and an f-unitary tensor $\mathscr{U} \in \mathbb{K}_n^{p\times s}$ such that $\mathscr{R} = \mathscr{U}^H * \mathscr{A} * \mathscr{U}$ contains the first s eigentubes on its diagonal.

```
for $k = 1, 2, \ldots$ until convergence do
    $\mathscr{X}_k = \mathscr{A}^q * \mathscr{X}_{k-1}$
    $\mathscr{X}_k = \mathscr{Q}_k * \mathscr{R}_k$
    $\mathscr{X}_k = \mathscr{Q}_k$
end for
$\mathscr{U} = \mathscr{X}_k$, $\mathscr{R} = \mathscr{U}^H * \mathscr{A} * \mathscr{U}$
```

The positive integer q in line 2 of Algorithm 6 determines the number of products with the tensor $\mathscr{A}$ in each iteration. A value $q > 1$ reduces the number of t-QR factorizations required to achieve a given power of $\mathscr{A}$ and, therefore, may speed up the computations. The t-QR factorization of the tensor $\mathscr{X}_k \in \mathbb{K}_n^{p\times s}$ is computed in line 3 of Algorithm 6. The frontal slices of the tensor $\mathscr{X}_k \in \mathbb{K}_n^{p\times s}$ converge to the desired t-Schur lateral slices under suitable conditions described in the following theorem.

Theorem 9. *Let the eigentubes $\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \ldots, \boldsymbol{\lambda}_p$ of the tensor $\mathscr{A} \in \mathbb{K}_n^{p\times p}$ satisfy*

$$\|\boldsymbol{\lambda}_1\| > \|\boldsymbol{\lambda}_2\| > \ldots > \|\boldsymbol{\lambda}_s\| > \|\boldsymbol{\lambda}_{s+1}\| \geqslant \|\boldsymbol{\lambda}_{s+2}\| \geqslant \ldots \geqslant \|\boldsymbol{\lambda}_p\|.$$

Introduce $\mathscr{Q} = [\vec{\mathscr{Q}}_1, \ldots, \vec{\mathscr{Q}}_s] \in \mathbb{K}_n^{p\times s}$, where $\vec{\mathscr{Q}}_i$ is the t-Schur lateral slice associated with $\boldsymbol{\lambda}_i$, $1 \leqslant i \leqslant s$, and let $\mathscr{P}_i$ denote the spectral projector-tensor associated with $\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_s$, and let the initial iterate be $\mathscr{X}_0 = [\vec{\mathscr{X}}_1, \ldots, \vec{\mathscr{X}}_s] \in \mathbb{K}_n^{p\times s}$. Assume that

$$\operatorname{rank}\left(\mathscr{P}_i * \left[\vec{\mathscr{X}}_1, \ldots, \vec{\mathscr{X}}_i\right]\right) = i, \quad i = 1, 2, \ldots, s.$$

Then the span$_{\mathrm{t}}$ of the i^{th} lateral slice $\vec{\mathscr{X}}_{k,i}$ of $\mathscr{X}_k$ generated by Algorithm 6 converges to span$_{\mathrm{t}}\{\vec{\mathscr{Q}}_i\}$.

Proof. The proof is analogous to the matrix case; see Saad [35, Theorem 5.1] for the latter. □

We remark that subspace iteration for matrices can be enhanced in a variety of ways, e.g., by including projections and deflation; see Saad [35, Theorem 5.1]. Algorithm 6 can be enhanced similarly.

5.6. The t-QR algorithm for tensors

The QR algorithm is the default method for computing all eigenpairs of a matrix of small to moderate size. A fairly recent discussion of the QR algorithm for matrices with references is provided in, e.g., [23]. This section describes an extension to the computation of eigenpairs of third-order tensors. This extension is referred to as the t-QR algorithm.

The default QR algorithm used for matrix computation is fairly complicated: it uses implicit shifts, and double shifts for real nonsymmetric matrices. In the interest of brevity, we only discuss a basic version of the t-QR algorithm. Details of a more efficient implementation that uses implicit and double shifts will be described elsewhere.

The unshifted t-QR algorithm described by Algorithm 7 is not applied in practice due to its slow convergence; it is also described in [25]. Our interest in this algorithm stems from that it is quite straightforward to show properties of the tensors it computes. Analogous properties can be shown for the tensors determined by the shifted algorithm discussed below.

Algorithm 7 The unshifted `t-QR` algorithm

```
Input: $\mathscr{A}_0 = \mathscr{A} \in \mathbb{K}_n^{p \times p}$.
Output: The eigentubes of $\mathscr{A}$.
1: for $k = 1$ until the convergence do
2:     Compute $\mathscr{A}_k = \mathscr{Q}_k * \mathscr{R}_k$
3:     Compute $\mathscr{A}_{k+1} = \mathscr{R}_k * \mathscr{Q}_k$
4: end for
```

Proposition 9. *Let $\mathscr{A} \in \mathbb{K}_n^{p \times p}$. Then the tensors $\mathscr{A}_1, \mathscr{A}_2, \ldots$ generated by Algorithm 7 satisfy*

$$\mathscr{A}_{k+1} = \left(\mathscr{Q}_1 * \mathscr{Q}_2 * \ldots * \mathscr{Q}_k\right)^H * \mathscr{A} * \left(\mathscr{Q}_1 * \mathscr{Q}_2 * \ldots * \mathscr{Q}_k\right). \tag{8}$$

Moreover,

$$\mathscr{A}^k = \left(\mathscr{Q}_1 * \mathscr{Q}_2 * \ldots * \mathscr{Q}_k\right) * \left(\mathscr{R}_k * \mathscr{R}_{k-1} * \ldots * \mathscr{R}_1\right), \tag{9}$$

where the sequences $\{\mathscr{Q}_i\}_{i=1}^k$ and $\{\mathscr{R}_i\}_{i=1}^k$ are defined by the t-QR factorization in line 2 of Algorithm 7.

Proof. Let $k \geqslant 1$. Then

$$\begin{aligned}
\mathscr{A}_{k+1} = \mathscr{R}_k * \mathscr{Q}_k &= \mathscr{Q}_k^H * \mathscr{Q}_k * \mathscr{R}_k * \mathscr{Q}_k = \mathscr{Q}_k^H * \mathscr{A}_k * \mathscr{Q}_k \\
&= \mathscr{Q}_k^H * \mathscr{R}_{k-1} * \mathscr{Q}_{k-1} * \mathscr{Q}_k \\
&= \mathscr{Q}_k^H * \mathscr{Q}_{k-1}^H * \mathscr{A}_{k-2} * \mathscr{Q}_{k-1} * \mathscr{Q}_k \\
&= \left(\mathscr{Q}_1 * \mathscr{Q}_2 * \ldots * \mathscr{Q}_k\right)^H \mathscr{A} \left(\mathscr{Q}_1 * \mathscr{Q}_2 * \ldots * \mathscr{Q}_k\right),
\end{aligned}$$

which gives (8). On the other hand,

$$\begin{aligned}
\mathscr{A}^k &= \mathscr{Q}_1 * \mathscr{R}_1 * \ldots * \mathscr{Q}_1 * \mathscr{R}_1 \\
&= \mathscr{Q}_1 * \mathscr{Q}_2 * \mathscr{R}_2 * \ldots * \mathscr{Q}_2 * \mathscr{R}_2 * \mathscr{R}_1 \\
&= \left(\mathscr{Q}_1 * \ldots * \mathscr{Q}_k\right) * \left(\mathscr{R}_k * \ldots * \mathscr{R}_1\right),
\end{aligned}$$

which shows (9). □

Eq. (8) shows that the tensor $\mathscr{A}_{k+1}$ is similar to $\mathscr{A}$, while relation (9) provides a `t-QR` factorization of the tensor $\mathscr{A}^k$. Letting

$$\breve{\mathscr{Q}}_k = \mathscr{Q}_1 * \mathscr{Q}_2 * \ldots * \mathscr{Q}_k \text{ and } \breve{\mathscr{R}}_k = \mathscr{R}_k * \mathscr{R}_{k-1} * \ldots * \mathscr{R}_1, \tag{10}$$

we obtain

$$\mathscr{A}_{k+1} = \breve{\mathscr{Q}}_k^H * \mathscr{A}_k * \breve{\mathscr{Q}}_k.$$

Thus, the tensor $\mathscr{A}_{k+1}$ is f-unitarily similar to $\mathscr{A}_k$.

The following result is concerned with the convergence of Algorithm 7. Convergence is considered under special conditions, but can be shown under less restrictive conditions as well.

Theorem 10. *Let* $\mathbb{K}_n^{p\times p}$ *be such that*

$$\mathscr{X}^{-1} * \mathscr{A} * \mathscr{X} = \mathbf{\Sigma} = t\text{- }diag\left[\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \ldots, \boldsymbol{\lambda}_p\right],$$

with

$$\|\boldsymbol{\lambda}_1\|_F > \|\boldsymbol{\lambda}_2\|_F > \ldots > \|\boldsymbol{\lambda}_p\|_F > 0, \tag{11}$$

and assume that all entries of every tube $\boldsymbol{\lambda}_k$ *are nonvanishing. Let* $\mathscr{X}^{-1} = \mathscr{L} * \mathscr{U}$ *denote the t-LU factorization of* $\mathscr{X}^{-1}$*, where* $\mathscr{L}$ *is a lower f-triangular tensor with the elements* $\mathbf{e}$ *on its diagonal; we assume for notational simplicity that pivoting is not required. Let* $\mathscr{X} = \mathscr{Q} * \mathscr{R}$ *be a t-QR factorization. Then there is an invertible f-diagonal tensor* $\mathscr{D}_k$*, such that* $\breve{\mathscr{Q}}_k * \mathscr{D}_k^{-1}$ *converges to* $\mathscr{Q}$ *as k increases, where* $\breve{\mathscr{Q}}_k$ *is given by* (10)*.*

Proof. We have

$$\begin{aligned}\mathscr{A}^k = \mathscr{X} * \mathbf{\Sigma}^k * \mathscr{X}^{-1} &= \mathscr{Q} * \mathscr{R} * \mathbf{\Sigma}^k * \mathscr{L} * \mathscr{U} \\ &= \mathscr{Q} * \mathscr{R} * \mathbf{\Sigma}^k * \mathscr{L} * \mathbf{\Sigma}^{-k} * \mathbf{\Sigma}^k * \mathscr{U},\end{aligned}$$

and we note that the elements of $\mathbf{\Sigma}^k * \mathscr{L} * \mathbf{\Sigma}^{-k}$ are given by

$$\mathbf{\Sigma}^k * \mathscr{L} * \mathbf{\Sigma}^{-k}(i,j,:) = \begin{cases} \mathscr{L}(i,j,:) * \left(\boldsymbol{\lambda}_i / \boldsymbol{\lambda}_j\right)^k & \text{if } i > j, \\ \mathbf{e} \text{ if } i = j, \\ \mathbf{0} \text{ otherwise.} \end{cases}$$

Then by using the ordering (11) of the eigentubes, the tensor $\mathbf{\Sigma}^k * \mathscr{L} * \mathbf{\Sigma}^{-k}$ converges to $\mathscr{I}$ as k increases. Thus, $\mathbf{\Sigma}^k * \mathscr{L} * \mathbf{\Sigma}^{-k}$ can be written as $\mathscr{I} + \mathscr{E}_k$, where $\mathscr{E}_k \to \mathbf{0}$ as k increases.

Since $\mathscr{X}$ is invertible, $\mathscr{R}$ also is invertible. Therefore

$$\begin{aligned}\mathscr{A}^k &= \mathscr{Q} * \mathscr{R} * (\mathscr{I} + \mathscr{E}_k) * \mathbf{\Sigma}^k * \mathscr{U} \\ &= \mathscr{Q} * \left(\mathscr{I} + \mathscr{R} * \mathscr{E}_k * \mathscr{R}^{-1}\right) * \mathscr{R} * \mathbf{\Sigma}^k * \mathscr{U}.\end{aligned}$$

Introduce the t-QR factorization $\mathscr{Q}'_k * \mathscr{R}'_k = \mathscr{I} + \mathscr{R} * \mathbf{\Sigma}_k * \mathscr{R}^{-1}$. Then

$$\mathscr{A}^k = \mathscr{Q} * \mathscr{Q}'_k * \mathscr{R}'_k * \mathscr{R} * \mathbf{\Sigma}^k * \mathscr{U}.$$

Let $\boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \ldots, \boldsymbol{\delta}_p$ denote the diagonal elements of the tensor $\mathscr{R}'_k * \mathscr{R} * \mathbf{\Sigma}^k * \mathscr{U}$. Since the elements on the diagonals of the tensors $\mathscr{R}'_k$, $\mathscr{R}$, $\mathbf{\Sigma}^k$, and $\mathscr{U}$ are nonvanishing, it follows that the tensor

$$\mathscr{D}_k = \text{t- diag}\left(\boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \ldots, \boldsymbol{\delta}_n\right) \in \mathbb{K}_n^{p\times p}$$

is invertible. We therefore can write

$$\mathscr{A}^k = \left(\mathscr{Q} * \mathscr{Q}'_k * \mathscr{D}_k\right) * \left(\mathscr{D}_k^{-1} * \mathscr{R}'_k * \mathscr{R} * \mathbf{\Sigma}^k * \mathscr{U}\right).$$

Then, by using the properties of the t-QR factorization, we conclude that

$$\breve{\mathscr{Q}}_k = \mathscr{Q} * \mathscr{Q}'_k * \mathscr{D}_k.$$

Moreover, since $\mathscr{Q}'_k$ converges to $\mathscr{I}$ as k increases, the tensor sequence

$$\breve{\mathscr{Q}}_k * \mathscr{D}_k^{-1} = \mathscr{Q} * \mathscr{Q}'_k, \quad k = 1, 2, \dots\ ,$$

converges to $\mathscr{Q}$.

Since the lateral slices of the tensor $\mathscr{Q}$ are the t-Schur lateral slices of $\mathscr{A}$, it follows that the scaled t-Schur lateral slices of $\mathscr{A}_k$ ($\breve{\mathscr{Q}}_k$) with elements in $\mathbb{K}_p$ converge to the lateral slices of $\mathscr{A}$ ($\mathscr{Q}$) as k increases. Consequently, $\mathscr{A}_k = \breve{\mathscr{Q}}_k^H * \mathscr{A} * \breve{\mathscr{Q}}_k$ converges to the upper f-triangular tensor related to the t-Schur lateral slices. □

The rate of convergence of Algorithm 7 can be accelerated by introducing shifts, i.e., by replacing lines 2 and 3 of the algorithm by $\mathscr{A}_k - \mathbf{\Sigma}_k * \mathscr{I}_p = \mathscr{Q}_k * \mathscr{R}_k$ and $\mathscr{A}_{k+1} = \mathscr{R}_k * \mathscr{Q}_k + \mathbf{\Sigma}_k$, respectively, where $\mathbf{\Sigma}_k = \text{t-diag}\,[\boldsymbol{\sigma}_1, \dots, \boldsymbol{\sigma}_k] \in \mathbb{K}_n^{p \times p}$ for suitable tubal shifts $\boldsymbol{\sigma}_k$. Moreover, the computational cost of each iteration can be reduced by first transforming $\mathscr{A}_0$ to f-upper-Hessenberg form. These modifications are described by Algorithm 8.

Algorithm 8 A shifted t-QR algorithm with f-Hessenberg reduction

Input: $\mathscr{A}_0 = \mathscr{A} \in \mathbb{K}_n^{p \times p}$, a shift tube $\boldsymbol{\sigma} \in \mathbb{K}_n$, $r = p$, $\epsilon > 0$. Define ($\mathbf{\Sigma} = \text{t-diag}\,[\boldsymbol{\sigma}, \dots, \boldsymbol{\sigma}] \in \mathbb{K}_n^{p \times p}$).

Output: The eigentubes of $\mathscr{A}$.

```
$[\mathscr{P}, \mathscr{H}_0] = \mathtt{t-hess}(\mathscr{A}_0)$ with $\mathscr{P} * \mathscr{H}_0 * \mathscr{P}^H = \mathscr{A}_0$
for $r > 1$ until convergence do
    $\mathscr{H}_k - \mathbf{\Sigma} = \mathscr{Q}_k * \mathscr{R}_k$   (Compute QR factorization)
    $\mathscr{A}_{k+1} = \mathscr{R}_k * \mathscr{Q}_* + \mathbf{\Sigma}$
    if $\|\mathscr{A}_{k+1}(r, r-1, :)\|_F \leqslant \epsilon$ then
        $r = r - 1$
    end if
end for
```

One way to determine suitable shifts $\boldsymbol{\sigma}$ is to let the shift in each iteration be $\mathscr{A}_k(r, r, :)$. We should mention that sometimes we need to reorder the tubes $\mathscr{A}_{k+1}(j, j-1, :)$, $j = 2, 3, \dots, r$, determined in each iteration of Algorithm 8 before determining the shift. This issue will be discussed in detail elsewhere.

6. Numerical experiments

This section illustrates the performance of the algorithms discussed in the previous section. The purpose of the examples is to demonstrate that the methods work similarly as the analogous methods for matrices. We used the following tensors in our experiments: The first tensor is of size 10 × 10 × 3 and has the frontal slices

$$\mathscr{A}(:,:,i) = \delta_i \, \texttt{tridiag}(-1, 2, -1),$$

with $\delta_i = 10^{i-1}$ for $1 \leqslant i \leqslant 3$. The second tensor in our experiments is the stochastic tensor $\mathscr{C} \in \mathbb{K}_4^{4\times 4}$ with frontal slices

$$\mathscr{C}(:,:,1) = \begin{pmatrix} 0.2091 & 0.2834 & 0.2194 & 0.1830 \\ 0.3371 & 0.3997 & 0.3219 & 0.3377 \\ 0.3265 & 0.0560 & 0.3119 & 0.2961 \\ 0.1273 & 0.2608 & 0.1468 & 0.1832 \end{pmatrix}, \quad \mathscr{C}(:,:,2) = \begin{pmatrix} 0.1952 & 0.2695 & 0.2055 & 0.1690 \\ 0.3336 & 0.3962 & 0.3184 & 0.3342 \\ 0.2954 & 0.0249 & 0.2808 & 0.2650 \\ 0.1758 & 0.3094 & 0.1953 & 0.2318 \end{pmatrix},$$

$$\mathscr{C}(:,:,3) = \begin{pmatrix} 0.3145 & 0.3887 & 0.3248 & 0.2883 \\ 0.0603 & 0.1230 & 0.0451 & 0.0609 \\ 0.3960 & 0.1255 & 0.3814 & 0.3656 \\ 0.2293 & 0.3628 & 0.2487 & 0.2852 \end{pmatrix}, \quad \mathscr{C}(:,:,4) = \begin{pmatrix} 0.1686 & 0.2429 & 0.1789 & 0.1425 \\ 0.3553 & 0.4180 & 0.3402 & 0.3559 \\ 0.3189 & 0.0484 & 0.3043 & 0.2885 \\ 0.1571 & 0.2907 & 0.1766 & 0.2131 \end{pmatrix}.$$

We also used the tensor $\texttt{complex}(10, 10, 10) \in \mathbb{K}_{10}^{10\times 10}$, whose entries have normally distributed real and imaginary parts with mean zero and variance one, as well as the real tensor $\texttt{real}(10, 10, 10)$, whose entries are normally distributed with mean zero and variance one, and with real eigentubes. Table 1 lists the abbreviations of the algorithms used in the experiments.

Method	Abbreviation	Algorithm
t-power method	t-PM	Algorithm 3
shifted inverse t-power method	t-SIPM	Algorithm 5
t-deflation method	t-DM	
t-subspace iteration	t-SI	Algorithm 6
t-QR algorithm with f-Hessenberg reduction and shifts	t-QRHS	Algorithm 8

Table 1: Abbreviations for the algorithms used in the experiments.

Let the diagonal entries of the f-diagonal tensor $\mathscr{D}_k \in \mathbb{K}_n^{k\times k}$ be the exact eigentubes of the tensors considered, and let the diagonal entries of the f-diagonal tensor $\widetilde{\mathscr{D}_k} \in \mathbb{K}_n^{k\times k}$ contain the corresponding approximations computed by the methods. We report the error

$$\text{Error} = \|\widetilde{\mathscr{D}_k} - \mathscr{D}_k\|_F. \tag{12}$$

All computations were carried out on a laptop computer with an 2.3 GHz Intel Core i5 processor and 8 GB of memory using MATLAB 2018a.

6.1. The tensor t-power method and the inverse t-power method

This subsection shows the performance of the t-power method and the closely related inverse t-power method. These methods are implemented by Algorithms 3 and 5. The t-power method determines an approximation of the eigenpair with the eigentube of largest norm, while the inverse t-power method computes an approximation of the eigenpair with the eigentube closest to a specified tubal shift.

We first consider the t-power method. Generically, Algorithms 3 determines a sequence $\{\vec{\mathscr{V}_k}\}_{k\geqslant 1}$ of approximations of an eigenslice that is associated with the eigentube of largest norm, as well of the sequence $\{\boldsymbol{\alpha}_k\}_{k\geqslant 1}$ of approximations of the largest eigentube; see Theorem 7. We will use the stopping criterion

$$\|\vec{\mathscr{V}}_{k+1} - \vec{\mathscr{V}}_k\|_F \leqslant \text{tol}, \text{ and } \|\boldsymbol{\alpha}_{k+1} - \boldsymbol{\alpha}_k\|_F \leqslant \text{tol},$$

where we set tol $= 10^{-15}$ in our tests and allow at most Itermax $= 3000$ iterations. The initial lateral slice $\vec{\mathscr{V}}_0$ is chosen to have normally distributed entries; the entries are chosen to be real if the tensor is real and the desired eigenslice is real; otherwise $\vec{\mathscr{V}}_0$ is complex with entries that have normally distributed real and imaginary parts. Let $\{\vec{\mathscr{U}}, \boldsymbol{\lambda}\}$ denote the computed approximation of the desired eigenpair of a tensor $\mathscr{F} \in \mathbb{K}_n^{p\times p}$. Then we evaluate the residual error norm

$$\text{Res.norm} = \|\mathscr{F} * \vec{\mathscr{U}} - \vec{\mathscr{U}} * \boldsymbol{\lambda}\|_F.$$

Table 2 reports some computed results. As can be expected, the t-power method may require many iterations to satisfy the convergence criterion.

Tensor	Method	Res.norm	Error	Iter	CPU time
$\mathscr{A}$	t-PM	2.14e-15	2.27e-15	537	0.382
$\mathscr{C}$	t-PM	3.70e-14	1.06e-14	2035	0.886
$\texttt{complex}(10,10,10)$	t-PM	3.47e-14	2.56e-14	849	0.707

Table 2: Residual norm, Error, number of iterations, and CPU time for t-PM.

As mentioned in Section 5.2, the power methods described by Algorithm 3 and by [26, Algorithm 4] apply different normalizations of the lateral tensor slices $\vec{\mathscr{V}}_k$. Table 3 and Figure 1 illustrates the performance of these algorithms when applied to the tensor $\mathscr{A}$ with the same initial tensor slice $\vec{\mathscr{V}}_0$.

k	$\|\vec{\mathscr{V}}_k - \vec{\mathscr{V}}_{k-1}\|_F$	
	Alg. 3	Alg. 4 in [26]
20	5.42e-02	7.04e-02
40	4.98e-03	7.12e-02
100	8.63e-05	6.71e-02
200	4.10e-07	6.71e-02
400	3.26e-12	6.71e-02

Table 3: Norms $\|\vec{\mathscr{V}}_k - \vec{\mathscr{V}}_{k-1}\|_F$ of the differences between successive lateral tensor slices generated by the power methods Algorithm 3 and [26, Algorithm 4] at iterations k.

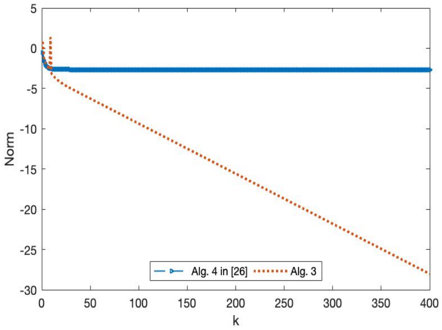

Figure 1: Plots of $\|\vec{\mathscr{V}}_k - \vec{\mathscr{V}}_{k-1}\|_F$ as a function of k for lateral tensor slices generated by Algorithm 3 (Alg. 3) and [26, Algorithm 4] (Alg. 4 in [26]) for $1 \leqslant k \leqslant 400$.

Theorem 6 shows that the power methods implemented by Algorithm 3 and [26, Algorithm 4] yield convergence when the initial lateral tensor slice $\vec{\mathscr{V}}_0$ is close enough to normalized eigenslice associated with the largest eigentube $\boldsymbol{\lambda}_1$ of a tensor $\mathscr{A}$. Table 3 and Figure 1 illustrate that what is "close enough" depends on the normalization of the lateral tensor slices $\vec{\mathscr{V}}_k$, $k = 1, 2, \dots$.

The inverse t-power method is implemented by Algorithm 5 with a stopping criterion that is analogous to the one for Algorithm 3. Here we seek to determine the eigenslice associated with the eigentube closest to the tubal shift $\boldsymbol{\sigma}$. The shifts $\boldsymbol{\sigma}$ used for the tensors $\mathscr{A}$ and `complex(10, 10, 10)` are the tube with $\boldsymbol{\sigma}(1,1,1) = 10^{-5}$ and remaining entries zero, and the tube with $\boldsymbol{\sigma}(1,1,1) = 10^{-3}$ and remaining entries zero, respectively; see Table 4.

Tensor	Method	Res. norm	Error	Iter
$\mathscr{A}$	t-SIPM	3.81e-15	2.91e-16	32
`complex(10, 10, 10)`	t-SIPM	4.33e-16	6.65e-15	422

Table 4: Residual norm, Error, and number of iterations for t-SIPM.

6.2. Deflation

We examine the performance of the previously discussed deflation techniques in order to approximate more than one eigenpair. Recall that the deflation methods differ in how we choose the lateral slice $\vec{\mathscr{V}}$; see (7). We choose $\vec{\mathscr{V}}$ as the i^{th} eigenslice (DE), as the i^{th} t-Schur lateral slice (DS), or as the i^{th} left eigenslice (DLE), and we report the results using the DE, DS, and DLE methods for the tensors $\mathscr{A}$ and `real` (10, 10, 10). To approximate the $(i+1)^{th}$ eigentube, with $i \geqslant 1$, we apply the power method to the new tensor

$$\mathscr{A}_{i+1} = \mathscr{A}_i - \vec{\mathscr{U}}_i * \boldsymbol{\lambda}_i * \vec{\mathscr{V}}_i^H,$$

where $\mathscr{A}_1 = \mathscr{A}$, $\boldsymbol{\lambda}_i$ is the previous approximated eigentube and $\vec{\mathscr{V}}_i$ is chosen as the previously approximated eigenslice (for the DE method) or as the left eigenslice (for the DLE method). We choose

$$\mathscr{A}_{i+1} = \mathscr{A}_i - \vec{\mathscr{Q}}_i * \boldsymbol{\lambda}_i * \vec{\mathscr{Q}}_i^H,$$

where $\vec{\mathscr{Q}}_i$ is the i^{th} t-Schur lateral slice, when applying the DS method. We approximate the first k eigenpairs of a given tensor $\mathscr{F} \in \mathbb{K}_n^{p\times p}$ and evaluate the residual norm

$$\text{Res.norm} = \|\mathscr{F} * \mathscr{U}_k - \mathscr{U}_k * \mathscr{D}_k\|_F,$$

where $\mathscr{U}_k \in \mathbb{K}_n^{p\times k}$ contains computed approximations of the eigenslices associated with the first eigentubes of $\mathscr{F}$, and $\mathscr{D}_k \in \mathbb{K}_n^{k\times k}$ is an f-diagonal tensor which on its diagonal has the computed approximations of the first k eigentubes of $\mathscr{F}$.

Table 5 reports the Error and the residual norm when approximating several eigentubes of the tensors $\mathscr{A}$ and $\texttt{real}(10, 10, 10)$ by the DE, DS, and DLE methods. Our aim is to approximate Num>1 eigentubes of the tensor.

Tensor	Num	DE			DLE			DS		
		Error	Res. norm	time	Error	Res. norm	time	Error	Res. norm	time
$\mathscr{A}$	3	4.58e-15	7.13e-15	0.675	4.19e-15	7.51e-15	1.237	4.57e-15	3.27e-15	0.668
	5	4.79e-15	6.87e-15	0.975	4.58e-15	7.63e-15	1.408	4.83e-15	3.43e-15	0.840
$\texttt{real}(10, 10, 10)$	4	8.06e-13	2.44e-13	0.642	8.09e-13	2.63e-13	1.246	8.19e-13	2.54e-13	0.641
	6	8.10e-13	2.98e-13	1.279	8.30e-13	2.91e-13	1.955	8.21e-13	3.08e-13	1.018

Table 5: The Error, the residual norm, and the CPU time for DE, DS, and DLE methods, when applied to the tensors $\mathscr{A}$ and $\texttt{real}\,(10, 10, 10)$ to approximate Num eigentubes.

Table 5 shows deflation with DS shifts to be faster than deflation with DE or DLE shifts. Figure 2 illustrates the convergence behavior for each eigenpair. We can see that the number of iterations required is the same for each eigenpair for all shifts DE, DEL, and DS.

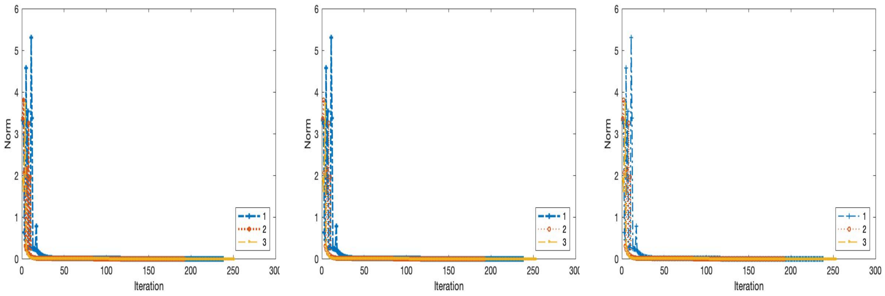

Figure 2: The behavior of the residual norm when we seek to approximate the first three eigenpairs of the tensor $\texttt{real}\,(10, 10, 10)$, by the DE, DLE, and DS methods from left to right.

6.3. Tensor subspace iteration

This subsection illustrates the performance of the t-subspace iteration method when applied to approximate eigenslices associated with the m eigentubes of largest norm of a tensor $\mathscr{A} \in \mathbb{K}_n^{p\times p}$. The sequence of approximate eigenslices

$\{\mathscr{X}_k\}_{k\geqslant 1}$ generated by Algorithm 6 generically converges to the/ span_{t} of the t-Schur lateral slices associated with the m largest eigentubes. Thus, the tensors

$$\mathscr{R}_k = \mathscr{X}_k^H * \mathscr{A} * \mathscr{X}_k, \quad k = 1, 2, \dots ,$$

converge to an f-upper triangular tensor. This suggests the convergence criterion

$$\text{error}(k) = \texttt{t-tril}\,(\mathscr{R}_{k+1} - \mathscr{R}_k) \leqslant \text{tol}, \tag{13}$$

where the function `t-tril` returns the f-lower triangular portion of its tensor argument. The tolerance used in all the experiments with the t-subspace method is $\text{tol} = 10^{-15}$, and the maximum number of iterations is set to $\text{Itermax} = 3000$. The initial tensor $\mathscr{X}_0$ has random entries chosen similarly as for the experiments with the t-power method.

To approximate the first m eigentubes, we compute the residual norm associated with the third-order tensor $\mathscr{F} \in \mathbb{K}_n^{p\times p}$,

$$\text{Res.norm} = \|\mathscr{F} * \mathscr{U}_m - \mathscr{U}_m * \mathscr{R}_m\|_F, \tag{14}$$

where $\mathscr{U}_m \in \mathbb{K}_n^{p\times m}$ is made up of computed approximations the t-Schur lateral slices associated the m largest eigentubes of $\mathscr{F}$, and $\mathscr{R}_m \in \mathbb{K}_n^{m\times m}$ is an upper (or quasi-upper) f-triangular tensor, whose f-diagonal entries are approximations of the m largest eigentubes of $\mathscr{F}$.

Table 6 displays the computed results. We notice that t-MSI is the same as t-SI when the power index q is 1. Figure 3 shows plots of the error (13) and of the residual norm at each iteration when seeking to approximate the four first eigentubes of the tensor $\texttt{complex}(10, 10, 10)$ for the power indices $q \in \{1, 4, 6\}$. Table 6 indicates that increasing the power index q has no effect on the accuracy, but reduces the total number of iterations.

Tensor	q	Error	Res.norm	Iter	CPU time
$\mathscr{A}$	1	4.58e-15	2.62e-15	490	0.288
	4	4.58e-15	2.36e-15	129	0.137
$\texttt{complex}(10, 10, 10)$	1	7.40e-11	2.10e-14	3000	2.775
	4	9.45e-14	2.40e-14	956	1.090

Table 6: The values of the Error, the residual norm, the number of iterations, and the CPU time for different values of the power index q obtained when using t-MSI for computing the first four eigentubes.

Figure 3 depicts the error norm and the residual norm for the t-MSI algorithm for the values 1, 4, 6 of the power index q. The results show fast convergence.

6.4. The t-QR algorithm

This subsection considers the t-QR algorithm t-QRHS. The algorithm is applied to the tensors $\mathscr{A}$ and $\mathscr{C}$. The tolerance is set to machine epsilon, i.e.,

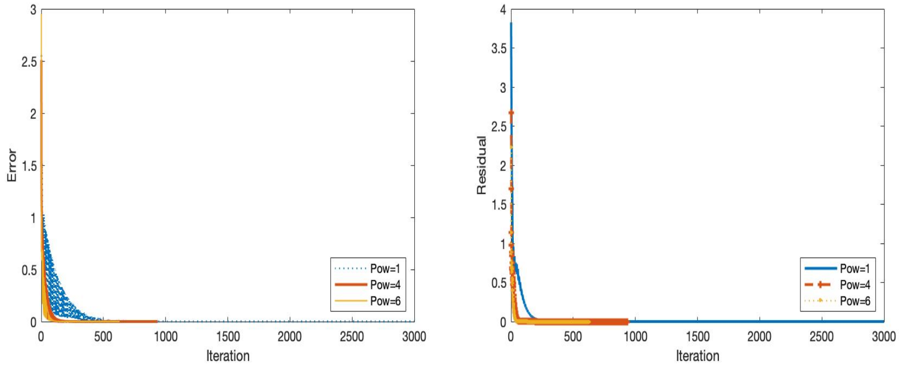

Figure 3: The curves representing the evolution of the error and the residual norm for algorithm t-MSI with different values of the power index $q \in \{1, 4, 6\}$ applied to the tensor `complex`$(10, 10, 10)$, when we desire approximations of the first four eigentubes.

$2.2204\ 10^{-16}$; the maximum number of iterations allowed for Algorithm8 is Itermax = 30000. We compute the residual norm defined by expression (14).

Table 7 reports the error norm (12), the residual norm, the CPU time, and the number of iterations required to satisfy the stopping criterion of the t-QRHS algorithm. The shifts have already been discussed after Algorithm 8, but since the last two eigentubes of the tensor $\mathscr{C}$ are complex, we used in each iteration the shift $\boldsymbol{\sigma} = \mathscr{H}_k(r, r, :) + i\mathscr{H}_k(r, r, :)$ for this tensor.

Tensor	Method	Error	Res.norm	CPU time	Iter
$\mathscr{A}$	t-QRHS	1.3984e-14	1.5373e-14	0.169	61
$\mathscr{C}$	t-QRHS	9.0322e-15	4.5962e-15	0.232	128

Table 7: The Error, the residual norm, the CPU time, and the number of iterations needed for convergence of t-QRHS when applied to the tensors $\mathscr{A}$ and $\mathscr{C}$.

7. Conclusion

This work discusses generalizations of eigenvalues and eigenvectors, referred to as eigentubes and eigenslices, respectively, for third-order tensors using the t-product. These eigentubes and eigenslices have previously been described by Kilmer et al. [25, 26]. We show new properties and describe several methods for computing eigentubes and eigenslices including the t-power method, the t-inverse iteration method, and t-subspace iteration, as well as the t-QR algorithm. We also introduce deflation techniques. Numerical tests illustrate the performance of these methods.

Acknowledgment

The authors would like to thank the referees for comments that lead to an improved presentation.

Conflict of Interest

The authors declare that they have no conflict of interest.

References

[1] S. Aeron, E. Kernfeld, M. Kilmer, Tensor-tensor products with invertible linear transforms, Linear Algebra Appl., 485, 545–570 (2015).

[2] H. Avron, L. Horesh, M. E. Kilmer, E. Newman, Tensor-tensor algebra for optimal representation and compression of multiway data, Proceedings of the National Academy of Sciences, National Acad Sciences, 188, 28 e2015851118 (2021).

[3] B. W. Bader, T. Kolda, Tensor decompositions and applications, SIAM Rev., 51, 455–500 (2009).

[4] F. P. A. Beik, M. Najafi-Kalyani, L. Reichel, Iterative Tikhonov regularization of tensor equations based on the Arnoldi process and some of its generalizations, Appl. Numer. Math., 151, 425–447 (2020).

[5] A. H. Bentbib, M. Boubekraoui, K. Jbilou, Vector Aitken extrapolation method for multilinear PageRank computations, J. Appl. Math. Comput., 1–28 (2022).

[6] A. H. Bentbib, A. El Hachimi, K. Jbilou, A. Ratnani, A tensor regularized nuclear norm method for image and video completion, J. Optim. Theory Appl., 192, 401–425, (2022).

[7] A. H. Bentbib, A. El Hachimi, K. Jbilou, A. Ratnani, Fast multidimensional completion and principal component analysis methods via the cosine product, Calcolo, 59, 1–33, (2022).

[8] A. H. Bentbib, A. Khouia, H. Sadok, The LSQR method for solving tensor least-squares problems, Electron. Trans. Numer. Anal., 55, 92–111, (2022).

[9] A. H. Bentbib, A. Khouia, H. Sadok, Color image and video restoration using tensor CP decomposition, BIT Numer. Math., 62, 1257–1278 (2022);

[10] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, Comput. Networks ISDN Systems, 30, 107–117 (1998).

[11] C. Bu, L. Sun, Y. Wei, B. Zheng, Moore–Penrose inverse of tensors via Einstein product, Linear Multilinear Algebra, 64, 686–698 (2016).

[12] Z. Chen, C. Li, S. Li, J. Zhao, Eigenvalue bounds of third-order tensors via the minimax eigenvalue of symmetric matrices, Comput. Appl. Math., 39, 1–14 (2020).

[13] S. Cipolla, M. Redivo-Zaglia, F. Tudisco, Shifted and extrapolated power methods for tensor ℓ^p-eigenpairs, Electron. Trans. Numer. Anal., 53, 1–27 (2020).

[14] M. Ding, T. -Z. Huang, T. -Y. Ji, X. -L. Zhao, J. -H. Yang, Low-rank tensor completion using matrix factorization based on tensor train rank and total variation, J. Sci. Comput., 81, 941–964 (2019).

[15] F. Dufrenois, A. El Ichi, K. Jbilou, Multilinear discriminant analysis using tensor-tensor products, J. Math. Modeling, in press.

[16] M. El Guide, A. El Ichi, K. Jbilou, R. Sadaka. On tensor GMRES and Golub-Kahan methods via the t-product for color image processing, Electron. J. Linear Algebra, 37, 524–543 (2021).

[17] M. El Guide, A. El Ichi, K. Jbilou, F. P. A. Beik, Tensor Krylov subspace methods via the Einstein product with applications to image and video processing, Appl. Numer. Math., 181, 347–363 (2022).

[18] A. El Hachimi, K. Jbilou, A. Ratnani, L. Reichel, A tensor bidiagonalisation method for higher-order singular value decomposition with applications, submitted for publication.

[19] S. El-Halouy, S. Noschese, L. Reichel, Perron communicability and sensitivity of multilayer networks, Numer. Algorithms, 92, 597–617 (2023).

[20] M. El Guide, A. El Ichi, K. Jbilou, Discrete cosine transform LSQR methods for multidimensional ill-posed problems, J. Math. Model., 10, 21–37 (2022).

[21] D. F. Gleich, L.-H. Lim, Y. Yu, Multilinear pagerank, SIAM J. Matrix Anal. Appl., 36, 1507–1541 (2015).

[22] G. H. Golub, H. A. Van der Vorst, Eigenvalue computation in the 20th century, J. Comput. Appl. Math., 123, 35–65, (2000).

[23] G. H. Golub, C. F. Van Loan, Matrix Computations, 4th ed., Johns Hopkins University Press, Baltimore, 2013.

[24] P. Henrici, Applied and Computational Complex Analysis, vol. 3, Wiley, New York, 1986.

[25] M. E. Kilmer, K. Braman, N. Hao, Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging, Report 2011,
http://www.cs.tufts.edu/t/tech_reports/reports/2011-01/report.pdf

[26] M. E. Kilmer, K. Braman, N. Hao, R. C. Hoover, Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging, SIAM J. Matrix Anal. Appl., 34, 148–172 (2013).

[27] M. E. Kilmer, C. D. Martin, Factorization strategies for third-order tensors, Linear Algebra Appl., 435, 641—658 (2011).

[28] R. Lehoucq, J. A. Scott, An evaluation of subspace iteration software for sparse nonsymmetric eigenproblems, SCAN-9605011, (1996).

[29] C. D. Martin, R. Shafer, B. LaRue, An order-p tensor factorization with applications in imaging, SIAM J. Sci. Comput., 35, A474–A490 (2013).

[30] L. Qi, Z. Luo, Tensor Analysis: Spectral Theory and Special Tensors, SIAM, Philadelphia, 2017.

[31] L. Reichel, U. O. Ugwu, Tensor Arnoldi-Tikhonov and GMRES-type methods for ill-posed problems with a t-product structure, J. Sci. Comput., 90, Art. 59, (2022).

[32] L. Reichel, U. O. Ugwu, The tensor Golub-Kahan-Tikhonov method applied to the solution of ill-posed problems with a t-product structure, Numer. Linear Algebra Appl., 29, Art e2412 (2022).

[33] L. Reichel, U. O. Ugwu, Weighted tensor Golub-Kahan-Tikhonov-type methods applied to image processing using a t-product, J. Comput. Appl. Math., 415, Art. 114488 (2022).

[34] H. Rojo, O. Rojo, Some results on symmetric circulant matrices and on symmetric centrosymmetric matrices, Linear Algebra Appl., 391, 211–233 (2004).

[35] Y. Saad, Numerical Methods for Large Eigenvalue Problems, 2nd ed., SIAM, Philadelphia, 2011.

[36] R. v. Mises, H. Pollaczek-Geiringer, Praktische Verfahren der Gleichungsauflösung, ZAMM, 9, 152–164 (1929).