NUMERICAL LINEAR ALGEBRA WITH APPLICATIONS
Numer. Linear Algebra Appl. 0000; **00**:1–22
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nla

The power and Arnoldi methods in an algebra of circulants

David F. Gleich[1*], Chen Greif[2], and James M. Varah[2]

[1] *Computer Science, Purdue University, West Lafayette, IN*
[2] *Computer Science, University of British Columbia, Vancouver, BC*

SUMMARY

Circulant matrices play a central role in a recently proposed formulation of three-way data computations. In this setting, a three-way table corresponds to a matrix where each "scalar" is a vector of parameters defining a circulant. This interpretation provides many generalizations of results from matrix or vector-space algebra. These results and algorithms are closely related to standard decoupling techniques on block-circulant matrices using the fast Fourier transform. We derive the power and Arnoldi methods in this algebra. In the course of our derivation, we define inner products, norms, and other notions. These extensions are straightforward in an algebraic sense, but the implications are dramatically different from the standard matrix case. For example, the number of eigenpairs may exceed the dimension of the matrix, though it is still polynomial in it. It is thus necessary to take an extra step and carefully select a smaller, canonical set of size equal to the dimension of the matrix from which all possible eigenpairs can be formed. Copyright © 0000 John Wiley & Sons, Ltd.

KEY WORDS: block-circulant, circulant module, tensor, FIR matrix algebra, power method, Arnoldi process

1. INTRODUCTION

We introduce and study iterative algorithms in a circulant algebra, which is a recent proposal by Kilmer et al. for a set of operations that generalize matrix algebra to three-way data [1, 2]. In particular, we extend this algebra with the ingredients required for iterative methods such as the power method and Arnoldi method, and study the behavior of these two algorithms.

Given an $m \times n \times k$ table of data, we view this data as an $m \times n$ matrix where each "scalar" is a vector of length k. We denote the space of length-k scalars as $\mathbb{K}_k$. These scalars interact like circulant matrices. Circulant matrices are a commutative, closed class under the standard matrix operations. Indeed, $\mathbb{K}_k$ is the ring of circulant matrices, where we identify each circulant matrix with the k parameters defining it.

Formally, let $\underline{\alpha} \in \mathbb{K}_k$. Elements in the circulant algebra are denoted by an underline to distinguish them from regular scalars. When an element is written with an explicit parameter set, it is denoted by braces, for example

$$\underline{\alpha} = \{\alpha_1 \ \cdots \ \alpha_k\}.$$

In what follows, we will use the notation $\leftrightarrow$ to provide an equivalent matrix-based notation for an operation involving $\mathbb{K}_k$. The following operations will define the basic computational

*Correspondence to: dgleich@purdue.edu

Contract/grant sponsor: This work was supported in part by the Natural Sciences and Engineering Research Council of Canada.

routines to treat $m \times n \times k$ arrays as $m \times n$ matrices of $\mathbb{K}_k$. They are equivalent to those proposed by Kilmer et al. [1, 2], and they constitute a module over vectors composed of circulants, as shown recently in Braman [3]. Our notation differs slightly. We define the operation $\texttt{circ}(\cdot)$ as the "circulant matrix representation" of a scalar:

$$\underline{\alpha} \quad \leftrightarrow \quad \texttt{circ}(\underline{\alpha}) \equiv \begin{bmatrix} \alpha_1 & \alpha_k & \dots & \alpha_2 \\ \alpha_2 & \alpha_1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \alpha_k \\ \alpha_k & \dots & \alpha_2 & \alpha_1 \end{bmatrix}. \tag{1}$$

Let $\underline{\alpha}$ be as above, and also let $\underline{\beta} \in \mathbb{K}_k$. The basic addition and multiplication operations between scalars are then

$$\underline{\alpha} + \underline{\beta} \leftrightarrow \texttt{circ}(\underline{\alpha}) + \texttt{circ}(\underline{\beta}) \quad \text{and} \quad \underline{\alpha} \circ \underline{\beta} \leftrightarrow \texttt{circ}(\underline{\alpha})\,\texttt{circ}(\underline{\beta}). \tag{2}$$

We use here a special symbol, the $\circ$ operation, to denote the product between these scalars, highlighting the difference from the standard matrix product. Note that the element

$$\underline{1} = \{1\ 0\ \dots\ 0\}$$

is the multiplicative identity.

Operations between vectors and matrices have similar, matricized, expressions. We use $\mathbb{K}_k^n$ to denote the space of length-n vectors where each component is a k-vector in $\mathbb{K}_k$, and $\mathbb{K}_k^{m\times n}$ to denote the space of $m \times n$ matrices of these k-vectors. Thus, we identify each $m \times n \times k$ table with an element of $\mathbb{K}_k^{m\times n}$. Let $\underline{\boldsymbol{A}} \in \mathbb{K}_k^{m\times n}$ and $\underline{\mathbf{x}} \in \mathbb{K}_k^n$. Their product is:

$$\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}} = \begin{bmatrix} \sum_{j=1}^n \underline{A}_{1,j} \circ \underline{x}_j \\ \vdots \\ \sum_{j=1}^n \underline{A}_{m,j} \circ \underline{x}_j \end{bmatrix} \leftrightarrow \begin{bmatrix} \texttt{circ}(\underline{A}_{1,1}) & \dots & \texttt{circ}(\underline{A}_{1,n}) \\ \vdots & \ddots & \vdots \\ \texttt{circ}(\underline{A}_{m,1}) & \dots & \texttt{circ}(\underline{A}_{m,n}) \end{bmatrix} \begin{bmatrix} \texttt{circ}(\underline{x}_1) \\ \vdots \\ \texttt{circ}(\underline{x}_n) \end{bmatrix}. \tag{3}$$

Thus, we extend the operation $\texttt{circ}$ to matrices and vectors of $\mathbb{K}_k$ scalars so that

$$\texttt{circ}(\underline{\boldsymbol{A}}) \equiv \begin{bmatrix} \texttt{circ}(\underline{A}_{1,1}) & \dots & \texttt{circ}(\underline{A}_{1,n}) \\ \vdots & \ddots & \vdots \\ \texttt{circ}(\underline{A}_{m,1}) & \dots & \texttt{circ}(\underline{A}_{m,n}) \end{bmatrix} \quad \text{and} \quad \texttt{circ}(\underline{\mathbf{x}}) \equiv \begin{bmatrix} \texttt{circ}(\underline{x}_1) \\ \vdots \\ \texttt{circ}(\underline{x}_n) \end{bmatrix}. \tag{4}$$

The definition of the product can now be compactly written as

$$\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}} \leftrightarrow \texttt{circ}(\underline{\boldsymbol{A}})\,\texttt{circ}(\underline{\mathbf{x}}). \tag{5}$$

Of course this notation also holds for the special case of scalar-vector multiplication. Let $\underline{\alpha} \in \mathbb{K}_k$. Then

$$\underline{\mathbf{x}} \circ \underline{\alpha} \leftrightarrow \texttt{circ}(\underline{\mathbf{x}})\,\texttt{circ}(\underline{\alpha}).$$

Based on this analysis, we term the set of operations the *circulant algebra*. We note that these operations have more efficient implementations, which will be discussed in Sections 3 and 6.

1.1. Contribution and outline

All prior algorithms for the circulant algebra have been direct methods. Our contribution in this paper is to derive the power and Arnoldi iterative methods. Towards this end, we first define inner products, norms, and other notions necessary for these algorithms (Section 2). We then continue this discussion by studying these same operations using the Fourier transform of the underlying circulant matrices (Section 3). In Fourier space, the matrix representation decouples, as illustrated in Figure 2. Theoretical properties of eigenvalues in the circulant algebra are analyzed next (Section 4). An important finding is that there may be more eigenpairs than the dimensionof the matrix, but these can all be presented by a unique

DOI: 10.1002/nla

canonical set (Theorem 7). Our discussion continues with the main contribution: the power method [4] and the Arnoldi method [5, 6, 7] in this algebra (Section 5). We next explain how we implemented these operations in a MATLAB package (Section 6); and we provide a numerical example of the algorithms (Section 7). Section 8 concludes the manuscript with some ideas for future work.

1.2. Related work

A recent preprint by Kilmer, Braman, and Hao [8] addresses many of the same issues. We use a slightly different notation and treat the notions of angle and ordering of elements in the algebra differently. A common theme in the study of this algebra is the use of the Fourier transform to decouple the problem. For instance, Kilmer and Martin [2] utilize this aspect to state the QR factorization in the algebra and Braman [3] uses it to study eigenvalues.

The circulant algebra analyzed in this paper is closely related to the *FIR matrix algebra* due to Lambert [9, Chapter 3]. Lambert proposes an algebra of circulants; but his circulants are padded with additional zeros to better approximate a finite impulse response operator. He uses it to study blind deconvolution problems [10]. As he observed, the relationship with matrices implies that many standard decompositions and techniques from real or complex valued matrix algebra carry over to the circulant algebra.

Furthermore, the circulant algebra is a particular instance of a matrix-over-a-ring, a long studied generalization of linear algebra [11, 12]. Prior work focuses on Roth theorems for the equation $AX - XB = C$ [13]; generalized inverses [14]; completion and controllability problems [15]; matrices over the ring of integers for computer algebra systems [16]; and transfer functions and linear dynamic systems [17]. Finally, see Gustafson [18] for some interesting relationships between vectors space theory and module theory. A recent proposal by Navasca et al. [19] extends many of Kilmer et al. [1] operations to more general algebraic structures.

More generally, multi-way arrays, tensors, and hypermatrices are a burgeoning area of research; see Kolda and Bader [20] for a recent comprehensive survey. Some of the major themes are multi-linear operations, fitting multi-linear models, and multi-linear generalizations of eigenvalues [21]. Savas and Eldén [22, 23] proposed a different generalization of Krylov methods for a tensor-product of vector-spaces and compute bases for each of the constituent spaces. This results in a tensor Krylov decomposition, which is fundamentally different from the standard matrix case. Even though our paper utilizes a Krylov method, namely the Arnoldi process, the similarity with Savas and Eldén is incidental as we treat a tensor as a map to a block-circulant matrix. In this sense, our paper is more closely related to Brazell et al. [24], who study the general setting of interpreting tensor data as a matrix. We study the problem when the map induces a circulant structure specifically and deal with some of the non-uniqueness issues that arise.

Our formulation is also closely related to block-circulant matrices, which have been studied for quite some time. See Tee [25] and the references therein for further historical and mathematical context on circulant matrices. In particular, Baker [26] gives a procedure for the SVD of a block circulant that involves using the fast Fourier transform to decouple the problem into independent sub-problems, just as we shall do throughout this manuscript. Other work in this vein includes solving block-circulant systems that arise in the theory of antenna arrays: [27, 28, 29]. Rezghi and Eldén [30] study circulant structure in tensors and how this can be diagonalized via an appropriate fast Fourier transform.

2. OPERATIONS WITH THE POWER METHOD

In the introduction, we provided the basic set of operations in the circulant algebra (eqs. (1)-(5)). We begin this section by stating the standard power method, and then follow by deriving the operations it requires.

Prepared using nlaauth.cls
DOI: 10.1002/nla

Figure 1. The power method for a matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$.

Require: $\boldsymbol{A}, \mathbf{x}^{(0)}, \tau$
1: $\mathbf{x}^{(0)} \leftarrow \mathbf{x}^{(0)} \left\|\mathbf{x}^{(0)}\right\|^{-1}$
2: **for** $k = 1, \ldots,$ until convergence **do**
3: $\quad \mathbf{y}^{(k)} \leftarrow \boldsymbol{A}\mathbf{x}^{(k-1)}$
4: $\quad \alpha^{(k)} \leftarrow \left\|\mathbf{y}^{(k)}\right\|$
5: $\quad \mathbf{x}^{(k)} \leftarrow \mathbf{y}^{(k)} {\alpha^{(k)}}^{-1}$
6: $\quad$ **if** $\|\text{sign}(x_1^{(k)})\mathbf{x}^{(k)} - \text{sign}(x_1^{(k-1)})\mathbf{x}^{(k-1)}\| < \tau$ **then**
7: $\quad\quad$ **return** $\mathbf{x}^{(k)}$
8: $\quad$ **end if**
9: **end for**

Let $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ and let $\mathbf{x} \in \mathbb{R}^n$ be an arbitrary starting vector. Then the power method proceeds by repeated applications of $\boldsymbol{A}$; see Figure 1 for a standard algorithmic description. (Line 6 checks for convergence and is one of several possible stopping criteria.) Under mild and well-known conditions (see Stewart [31]), this iteration converges to the eigenvector with the largest magnitude eigenvalue.

Not all of the operations in Figure 1 are defined for the circulant algebra. In the first line, we use the norm $\left\|\mathbf{x}^{(0)}\right\|$ that returns a scalar in $\mathbb{R}$. We also use the scalar inverse α^{-1}. The next operation is the sign function for a scalar. Let us define these operations, in order of their complexity. In the next section, we will reinterpret these operations in light of the relationships between the fast Fourier transform and circulant matrices. This will help illuminate a few additional properties of these operations and will let us state an ordering for elements.

2.1. The scalar inverse

We begin with the scalar inverse. Recall that all operations between scalars behave like circulant matrices. Thus, the inverse of $\underline{\alpha} \in \mathbb{K}_k$ is

$$\underline{\alpha}^{-1} \leftrightarrow \texttt{circ}(\underline{\alpha})^{-1}.$$

The matrix $\texttt{circ}(\underline{\alpha})^{-1}$ is also circulant [32]. If $\texttt{circ}(\underline{\alpha})$ is singular, then the inverse may not exist. This issue arises frequently in studies of the algebra [3, 8]. Our appoach in this event is to use the pseudo-inverse of the matrix, which would suffice for the power method and Arnoldi method addressed in this paper.

2.2. Scalar functions and the angle function

Other scalar functions are also functions of a matrix (see Higham [33]). Let f be a function, then

$$f(\underline{\alpha}) \leftrightarrow f(\texttt{circ}(\underline{\alpha}))$$

where the right hand side is the same function applied to a matrix. (Note that it is not the function applied to the matrix element-wise.)

The sign function for a matrix is a special case. As explained in Higham [33], the sign function applied to a complex value is the sign of the real-valued part. We wish to use a related concept that generalizes the real-valued sign that we term "angle." Given a complex value $re^{\imath\theta}$, then $\texttt{angle}(re^{\imath\theta}) = e^{\imath\theta}$. For real or complex numbers x, we then have

$$\texttt{angle}(x)\,|x| = x.$$

Thus, we define

$$\texttt{angle}(\underline{\alpha}) \leftrightarrow \texttt{circ}(\texttt{abs}(\underline{\alpha}))^{-1}\,\texttt{circ}(\underline{\alpha}).$$

DOI: 10.1002/nla

2.3. Inner products, norms, and conjugates

We now proceed to define a norm. The norm of a vector in $\mathbb{K}_k^n$ produces a scalar in $\mathbb{K}_k$:

$$\|\underline{\mathbf{x}}\| \leftrightarrow (\mathtt{circ}(\underline{\mathbf{x}})^* \, \mathtt{circ}(\underline{\mathbf{x}}))^{1/2} = \left(\sum_{i=1}^{n} \mathtt{circ}(\underline{x}_i)^* \, \mathtt{circ}(\underline{x}_i) \right)^{1/2}.$$

For a standard vector $\mathbf{x} \in \mathbb{C}^n$, the norm $\|\mathbf{x}\| = \sqrt{\mathbf{x}^*\mathbf{x}}$. This definition, in turn, follows from the standard inner product attached to the vector space $\mathbb{C}^n$. As we shall see, our definition has a similar interpretation. The inner product implied by our definition is

$$\langle \underline{\mathbf{x}}, \underline{\mathbf{y}} \rangle \leftrightarrow \mathtt{circ}(\underline{\mathbf{y}})^* \, \mathtt{circ}(\underline{\mathbf{x}}).$$

Additionally, this definition implies that that the *conjugate* operation in the circulant algebra corresponds to the transpose of the circulant matrix

$$\overline{\underline{\alpha}} \leftrightarrow \mathtt{circ}(\underline{\alpha})^*.$$

With this conjugate, our inner product satisfies two of the standard properties: conjugate symmetry $\langle \underline{\mathbf{x}}, \underline{\mathbf{y}} \rangle = \overline{\langle \underline{\mathbf{y}}, \underline{\mathbf{x}} \rangle}$ and linearity $\langle \underline{\alpha} \circ \underline{\mathbf{x}}, \underline{\mathbf{y}} \rangle = \underline{\alpha} \circ \langle \underline{\mathbf{x}}, \underline{\mathbf{y}} \rangle$. The notion of positive definiteness is more intricate and we delay that discussion until after introducing a decoupling technique using the fast Fourier transform in the following section. Then, in Section 3.3, we use positive definiteness to demonstrate a Cauchy-Schwarz inequality, which in turn provides a triangle inequality for the norm.

3. OPERATIONS WITH THE FAST FOURIER TRANSFORM

In Section 2, we explained the basic operations of the circulant algebra as operations between matrices. All of these matrices consisted of circulant blocks. In this section, we show how to accelerate these operations by exploiting the relationship between the fast Fourier transform and circulant matrices.

Let $\boldsymbol{C}$ be a $k \times k$ circulant matrix. Then the eigenvector matrix of $\boldsymbol{C}$ is given by the $k \times k$ discrete Fourier transform matrix $\boldsymbol{F}$, where

$$F_{ij} = \frac{1}{\sqrt{k}} \omega^{(i-1)(j-1)}$$

and $\omega = e^{2\pi\imath/k}$. This matrix is complex symmetric, $\boldsymbol{F}^T = \boldsymbol{F}$, and unitary, $\boldsymbol{F}^* = \boldsymbol{F}^{-1}$. Thus, $\boldsymbol{C} = \boldsymbol{F}\boldsymbol{D}\boldsymbol{F}^*$, $\boldsymbol{D} = \mathrm{diag}(\lambda_1, \ldots, \lambda_k)$. Recall that multiplying a vector by $\boldsymbol{F}$ or $\boldsymbol{F}^*$ can be accomplished via the fast Fourier transform in $\mathcal{O}(k \log k)$ time instead of $\mathcal{O}(k^2)$ for the typical matrix-vector product algorithm. Also, computing the matrix $\boldsymbol{D}$ can be done in time $O(k \log k)$ as well.

To express our operations, we define a new transformation, the "Circulant Fourier Transform" or `cft`. Formally, $\mathtt{cft} : \underline{\alpha} \in \mathbb{K}_k \mapsto \mathbb{C}^{k \times k}$ and its inverse $\mathtt{icft} : \mathbb{C}^{k \times k} \mapsto \mathbb{K}_k$ as follows:

$$\mathtt{cft}(\underline{\alpha}) \equiv \begin{bmatrix} \hat{\alpha}_1 & & \\ & \ddots & \\ & & \hat{\alpha}_k \end{bmatrix} = \boldsymbol{F}^* \, \mathtt{circ}(\underline{\alpha}) \boldsymbol{F}, \qquad \mathtt{icft}\left(\begin{bmatrix} \hat{\alpha}_1 & & \\ & \ddots & \\ & & \hat{\alpha}_k \end{bmatrix} \right) \equiv \underline{\alpha} \leftrightarrow \boldsymbol{F} \, \mathtt{cft}(\underline{\alpha}) \boldsymbol{F}^*,$$

where $\hat{\alpha}_j$ are the eigenvalues of $\mathtt{circ}(\underline{\alpha})$ as produced in the Fourier transform order. These transformations satisfy $\mathtt{icft}(\mathtt{cft}(\underline{\alpha})) = \underline{\alpha}$ and provide a convenient way of moving between operations in $\mathbb{K}_k$ to the more familiar environment of diagonal matrices in $\mathbb{C}^{k \times k}$.

The `cft` and `icft` transformations are extended to matrices and vectors over $\mathbb{K}_k$ differently than the `circ` operation we saw before. Observe that `cft` applied "element-wise" to the

DOI: 10.1002/nla

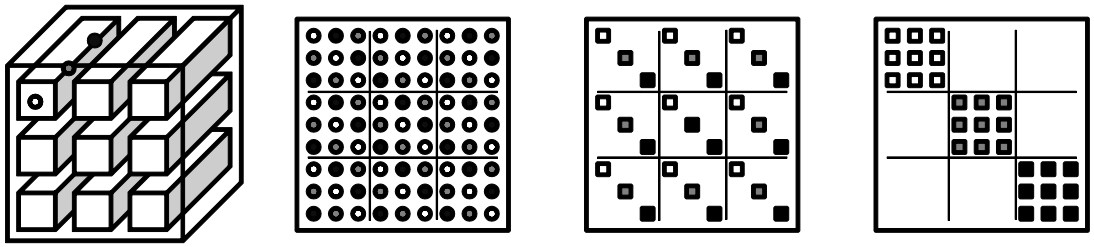

Figure 2. The sequence of transformations in our `cft` operation. Given a circulant $\boldsymbol{A}$, we convert it into a matrix by $\texttt{circ}(\boldsymbol{A})$. The color of the circles in the figure is emphasizing the circulant structure, and not equality between blocks. In the third figure, we diagonalize each circulant using the Fourier transform. The pattern of eigenvalues is represented by squares. Here, we are coloring the squares to show the reordering induced by the permutation at the final step of the `cft` operation.

$\texttt{circ}(\underline{\boldsymbol{A}})$ matrix produces a matrix of diagonal blocks. In our extension of the `cft` routine, we perform an additional permutation to expose block-diagonal structure from these diagonal blocks. This permutation $\boldsymbol{P}_m\boldsymbol{A}\boldsymbol{P}_n^T$ transforms an $mk\times nk$ matrix of $k\times k$ diagonal blocks into a block diagonal $mk\times nk$ with $m\times n$ size blocks. It is also known as a *stride permutation matrix* [34]. The construction of $\boldsymbol{P}_m$, expressed in MATLAB code is

```
p = reshape(1:m*k,k,m)';
Pm = sparse(1:m*k,p(:),1,m*k,m*k);
```

The construction for $\boldsymbol{P}_n$ is identical. In Figure 2, we illustrate the overall transformation process that extends `cft` to matrices and vectors.

Algebraically, the `cft` operation for a matrix $\underline{\boldsymbol{A}}\in\mathbb{K}_k^{m\times n}$ is

$$\texttt{cft}(\underline{\boldsymbol{A}}) = \boldsymbol{P}_m(\boldsymbol{I}_m\otimes\boldsymbol{F}^*)\,\texttt{circ}(\underline{\boldsymbol{A}})(\boldsymbol{I}_n\otimes\boldsymbol{F})\boldsymbol{P}_n^T,$$

where $\boldsymbol{P}_m$ and $\boldsymbol{P}_n$ are the permutation matrices introduced above. We can equivalently write this directly in terms of the eigenvalues of each of the circulant blocks of $\texttt{circ}(\underline{\boldsymbol{A}})$:

$$\texttt{cft}(\underline{\boldsymbol{A}}) \equiv \begin{bmatrix}\hat{\boldsymbol{A}}_1 & & \\ & \ddots & \\ & & \hat{\boldsymbol{A}}_k\end{bmatrix}, \qquad \hat{\boldsymbol{A}}_j = \begin{bmatrix}\lambda_j^{1,1} & \dots & \lambda_j^{1,n}\\ \vdots & \ddots & \vdots\\ \lambda_j^{m,1} & \dots & \lambda_j^{m,n}\end{bmatrix},$$

where $\lambda_1^{r,s},\dots,\lambda_k^{r,s}$ are the diagonal elements of $\texttt{cft}(\underline{A}_{r,s})$. The inverse operation `icft`, takes a block diagonal matrix and returns the matrix in $\mathbb{K}_k^{m\times n}$:

$$\texttt{icft}(\boldsymbol{A}) \leftrightarrow (\boldsymbol{I}_m\otimes\boldsymbol{F})\boldsymbol{P}_m^T\boldsymbol{A}\boldsymbol{P}_n(\boldsymbol{I}_n\otimes\boldsymbol{F}^*).$$

Let us close this discussion by providing a concrete example of this operation.

Example 1

Let $\underline{\boldsymbol{A}} = \begin{bmatrix}\{2\ 3\ 1\} & \{8\ -2\ 0\}\\ \{-2\ 0\ 2\} & \{3\ 1\ 1\}\end{bmatrix}$. The result of the `circ` and `cft` operations, as illustrated in Figure 2,

DOI: 10.1002/nla

are:

$$\texttt{circ}(\underline{\boldsymbol{A}}) = \left[\begin{array}{ccc|ccc} 2 & 1 & 3 & 8 & 0 & -2 \\ 3 & 2 & 1 & -2 & 8 & 0 \\ 1 & 3 & 2 & 0 & -2 & 8 \\ \hline -2 & 2 & 0 & 3 & 1 & 1 \\ 0 & -2 & 2 & 1 & 3 & 1 \\ 2 & 0 & -2 & 1 & 1 & 3 \end{array}\right],$$

$$(\boldsymbol{I} \otimes \boldsymbol{F}^*)\,\texttt{circ}(\underline{\boldsymbol{A}})(\boldsymbol{I} \otimes \boldsymbol{F}) = \left[\begin{array}{ccc|ccc} 6 & & & 6 & & \\ & -\sqrt{3}\imath & & & -9+\sqrt{3}\imath & \\ & & \sqrt{3}\imath & & & -9-\sqrt{3}\imath \\ \hline 0 & & & 5 & & \\ & -3+\sqrt{3}\imath & & & 2 & \\ & & -3-\sqrt{3}\imath & & & 2 \end{array}\right],$$

$$\texttt{cft}(\underline{\boldsymbol{A}}) = \left[\begin{array}{cc|cc|cc} 6 & 6 & & & & \\ 0 & 5 & & & & \\ \hline & & -\sqrt{3}\imath & -9+\sqrt{3}\imath & & \\ & & -3+\sqrt{3}\imath & 2 & & \\ \hline & & & & \sqrt{3}\imath & -9-\sqrt{3}\imath \\ & & & & -3-\sqrt{3}\imath & 2 \end{array}\right].$$

3.1. Operations

We now briefly illustrate how the `cft` accelerates and simplifies many operations. Let $\underline{\alpha}, \underline{\beta} \in \mathbb{K}_k$. Note that

$$\underline{\alpha} + \underline{\beta} = \texttt{icft}(\texttt{cft}(\underline{\alpha}) + \texttt{cft}(\underline{\beta})), \text{ and}$$
$$\underline{\alpha} \circ \underline{\beta} = \texttt{icft}(\texttt{cft}(\underline{\alpha})\,\texttt{cft}(\underline{\beta})).$$

In the Fourier space – the output of the `cft` operation – these operations are both $O(k)$ time because they occur between diagonal matrices. Due to the linearity of the `cft` operation, arbitrary sequences of operations in the Fourier space transform back seamlessly, for instance

$$\underbrace{(\underline{\alpha} + \underline{\beta}) \circ (\underline{\alpha} + \underline{\beta}) \circ \ldots \circ (\underline{\alpha} + \underline{\beta})}_{j \text{ times}} = \texttt{icft}((\texttt{cft}(\underline{\alpha}) + \texttt{cft}(\underline{\beta}))^j).$$

But even more importantly, these simplifications generalize to matrix-based operations too. For example,

$$\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}} = \texttt{icft}(\texttt{cft}(\underline{\boldsymbol{A}})\,\texttt{cft}(\underline{\mathbf{x}})).$$

In fact, in the Fourier space, this system is a series of independent matrix vector products:

$$\texttt{cft}(\underline{\boldsymbol{A}})\,\texttt{cft}(\underline{\mathbf{x}}) = \begin{bmatrix} \hat{\boldsymbol{A}}_1 & & \\ & \ddots & \\ & & \hat{\boldsymbol{A}}_k \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}_1 & & \\ & \ddots & \\ & & \hat{\mathbf{x}}_k \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{A}}_1\hat{\mathbf{x}}_1 & & \\ & \ddots & \\ & & \hat{\boldsymbol{A}}_k\hat{\mathbf{x}}_k \end{bmatrix}.$$

Here, we have again used $\hat{\boldsymbol{A}}_j$ and $\hat{\mathbf{x}}_j$ to denote the blocks of Fourier coefficients, or equivalently, circulant eigenvalues. *The rest of the paper frequently uses this convention and shorthand where it is clear from context.* This formulation takes

$$\underbrace{O(mnk \log k + nk \log k)}_{\texttt{cft and icft}} + \underbrace{O(kmn)}_{\text{matvecs}}$$

operations instead of $O(mnk^2)$ using the `circ` formulation in the previous section.

DOI: 10.1002/nla

More operations are simplified in the Fourier space too. Let $\texttt{cft}(\underline{\alpha}) = \text{diag}\,[\,\hat{\alpha}_1,\, ...,\, \hat{\alpha}_k\,]$. Because the $\hat{\alpha}_j$ values are the eigenvalues of $\texttt{circ}(\alpha)$, the following functions simplify:

$$\begin{aligned}\texttt{abs}(\underline{\alpha}) &= \texttt{icft}(\text{diag}\,[\,|\hat{\alpha}_1|,\, ...,\, |\hat{\alpha}_k|\,]),\\ \overline{\underline{\alpha}} &= \texttt{icft}(\text{diag}\,[\,\overline{\hat{\alpha}_1},\, ...,\, \overline{\hat{\alpha}_k}\,]) = \texttt{icft}(\texttt{cft}(\underline{\alpha})^*),\ \text{and}\\ \texttt{angle}(\underline{\alpha}) &= \texttt{icft}(\text{diag}\,[\,\hat{\alpha}_1/|\hat{\alpha}_1|,\, ...,\, \hat{\alpha}_k/|\hat{\alpha}_k|\,]).\end{aligned}$$

Complex values in the CFT. A small conern with the `icft` operation is that it may produce complex-valued elements in $\mathbb{K}_k$. Note that when the output of a sequence of circulant operations produces a real-valued circulant, then the output of `icft` is also real-valued. In other words, there is no problem working in Fourier space instead of the real-valued circulant space. This fact can be formally verified by first formally stating the conditions under which `icft` produces real-valued circulants ($\texttt{icft}(\boldsymbol{D})$ is real if and only if $\boldsymbol{F}^2\boldsymbol{D}\boldsymbol{F}^2 = \boldsymbol{D}^*$, see Davis [32]), and then checking that the operations in the Fourier space do not alter this condition.

3.2. Properties

Representations in Fourier space are convenient for illustrating some properties of these operations.

Proposition 2
The matrix $\texttt{circ}(\texttt{angle}(\underline{\alpha}))$ is orthogonal.

Proof
We have

$$\begin{aligned}&\texttt{circ}(\texttt{angle}(\underline{\alpha}))^*\,\texttt{circ}(\texttt{angle}(\underline{\alpha})) \leftrightarrow\\ &\qquad \overline{\texttt{angle}(\underline{\alpha})} \circ \texttt{angle}(\underline{\alpha}) = \texttt{icft}\left(\begin{bmatrix}\overline{\hat{\alpha}}_1\hat{\alpha}_1/|\hat{\alpha}_1|^2 & & \\ & \ddots & \\ & & \overline{\hat{\alpha}}_k\hat{\alpha}_k/|\hat{\alpha}_k|^2\end{bmatrix}\right) = \underline{1}.\end{aligned}$$ □

Additionally, the Fourier space is an easy place to understand spanning sets and bases in $\mathbb{K}_k^m$, as the following proposition shows.

Proposition 3
Let $\underline{\boldsymbol{X}} \in \mathbb{K}_k^{m\times n}$. Then $\underline{\boldsymbol{X}}$ spans $\mathbb{K}_k^m$ *if and only if* $\texttt{circ}(\underline{\boldsymbol{X}})$ and $\texttt{cft}(\underline{\boldsymbol{X}})$ have rank km. Also $\underline{\boldsymbol{X}}$ is a basis *if and only if* $\texttt{circ}(\underline{\boldsymbol{X}})$ and $\texttt{cft}(\underline{\boldsymbol{X}})$ are invertible.

Proof
First note that $\text{rank}(\texttt{cft}(\underline{\boldsymbol{X}})) = \text{rank}(\texttt{circ}(\underline{\boldsymbol{X}}))$ because `cft` is a similarity transformation applied to `circ`. It suffices to show this result for $\texttt{cft}(\underline{\boldsymbol{X}})$, then. Now consider $\underline{\mathbf{y}} = \underline{\boldsymbol{X}} \circ \underline{\mathbf{a}}$:

$$\begin{aligned}\texttt{cft}(\underline{\mathbf{y}}) &= \texttt{cft}(\underline{\boldsymbol{X}})\,\texttt{cft}(\underline{\mathbf{a}});\\ \begin{bmatrix}\hat{\mathbf{y}}_1 & & \\ & \ddots & \\ & & \hat{\mathbf{y}}_k\end{bmatrix} &= \begin{bmatrix}\hat{\boldsymbol{X}}_1 & & \\ & \ddots & \\ & & \hat{\boldsymbol{X}}_k\end{bmatrix}\begin{bmatrix}\hat{\mathbf{a}}_1 & & \\ & \ddots & \\ & & \hat{\mathbf{a}}_k\end{bmatrix}.\end{aligned}$$

Thus, if there is a $\underline{\mathbf{y}}$ that is feasible, then all $\hat{\boldsymbol{X}}_j \in \mathbb{C}^{m\times n}$ must be rank m. Conversely, if $\texttt{cft}(\underline{\boldsymbol{X}})$ has rank km then each $\hat{\boldsymbol{X}}_j$ must have rank m, and any $\underline{\mathbf{y}}$ is feasible. The result about the basis follows from an analogous argument. □

3.3. Inner products, norms, and ordering

We now return to our inner product and norm to elaborate on the positive-definiteness and the triangle inequality. In terms of the Fourier transform,

$$\langle \underline{\mathbf{x}}, \underline{\mathbf{y}} \rangle = \texttt{icft}(\texttt{cft}(\underline{\mathbf{y}})^*\,\texttt{cft}(\underline{\mathbf{x}})).$$

Prepared using nlaauth.cls
DOI: 10.1002/nla

If we write this in terms of the blocks of Fourier coefficients then

$$\mathtt{cft}(\underline{\mathbf{x}})^* \, \mathtt{cft}(\underline{\mathbf{y}}) = \begin{bmatrix} \hat{\mathbf{y}}_1^* \hat{\mathbf{x}}_1 & & \\ & \ddots & \\ & & \hat{\mathbf{y}}_k^* \hat{\mathbf{x}}_k \end{bmatrix}.$$

For $\underline{\mathbf{y}} = \underline{\mathbf{x}}$, each diagonal term has the form $\hat{\mathbf{x}}_j^* \hat{\mathbf{x}}_j \geq 0$. Consequently, we do consider this a positive semi-definite inner product because the output $\mathtt{circ}(\langle \underline{\mathbf{x}}, \underline{\mathbf{y}} \rangle)$ is a matrix with non-negative eigenvalues. This idea motivates the following definition of element ordering.

Definition 4 (Ordering)
Let $\underline{\alpha}, \underline{\beta} \in \mathbb{K}_k$. We write

$$\begin{array}{lll} \underline{\alpha} \leq \underline{\beta} & \text{when} & \operatorname{diag}(\mathtt{cft}(\underline{\alpha})) \leq \operatorname{diag}(\mathtt{cft}(\underline{\beta})) \quad \text{element-wise, and} \\ \underline{\alpha} < \underline{\beta} & \text{when} & \operatorname{diag}(\mathtt{cft}(\underline{\alpha})) < \operatorname{diag}(\mathtt{cft}(\underline{\beta})) \quad \text{element-wise.} \end{array}$$

We now show that our inner product satisfies the Cauchy-Schwarz inequality:

$$\mathtt{abs}\, \langle \underline{\mathbf{x}}, \underline{\mathbf{y}} \rangle \leq \|\underline{\mathbf{x}}\| \circ \left\|\underline{\mathbf{y}}\right\|.$$

In Fourier space, this fact holds because $|\hat{\mathbf{y}}_j^* \hat{\mathbf{x}}_j| \leq \|\hat{\mathbf{x}}_j\| \, \|\hat{\mathbf{y}}_j\|$ follows from the standard Cauchy-Schwarz inequality. Using this inequality, we find that our norm satisfies the triangle inequality:

$$\left\|\underline{\mathbf{x}} + \underline{\mathbf{y}}\right\|^2 = \langle \underline{\mathbf{x}} + \underline{\mathbf{y}}, \underline{\mathbf{x}} + \underline{\mathbf{y}} \rangle \leq \langle \underline{\mathbf{x}}, \underline{\mathbf{x}} \rangle + \underline{2} \circ \|\underline{\mathbf{x}}\| \circ \left\|\underline{\mathbf{y}}\right\| + \langle \underline{\mathbf{y}}, \underline{\mathbf{y}} \rangle = (\|\underline{\mathbf{x}}\| + \left\|\underline{\mathbf{y}}\right\|)^2.$$

In this expression, the constant $\underline{2}$ is twice the multiplicative identify, that is $\underline{2} = \{2 \; 0 \; \ldots \; 0\}$.

4. EIGENVALUES AND EIGENVECTORS

With the results of the previous few sections, we can now state and analyze an eigenvalue problem in circulant algebra. Braman [3] investigated these problems already and proposed a decomposition approach to compute the eigenvalues. We offer an extended analysis that addresses a few additional aspects. Specifically, we focus on a *canonical* set of eigenpairs.

Recall that eigenvalues of matrices are the roots of the characteristic polynomial $\det(\boldsymbol{A} - \lambda \boldsymbol{I}) = 0$. Now let $\underline{\boldsymbol{A}} \in \mathbb{K}_k^{n \times n}$ and $\underline{\lambda} \in \mathbb{K}_k$. The eigenvalue problem does not change:

$$\det(\underline{\boldsymbol{A}} - \underline{\lambda} \circ \underline{\boldsymbol{I}}) = \underline{0}.$$

(As an aside, note that the standard properties of the determinant hold for any matrix over a commutative ring with identity; in particular, the Cayley-Hamilton theorem holds in this algebra.) The existence of an eigenvalue implies the existence of a corresponding eigenvector $\underline{\mathbf{x}} \in \mathbb{K}_k^n$. Thus, an eigenvalue and eigenvector pair in this algebra is

$$\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}} = \underline{\lambda} \circ \underline{\mathbf{x}}.$$

Just like the matrix case, these eigenvectors can be rescaled by any constant $\underline{\alpha} \in \mathbb{K}_k$: $\underline{\boldsymbol{A}} \circ \underline{\alpha} \circ \underline{\mathbf{x}} = \underline{\lambda} \circ \underline{\alpha} \circ \underline{\mathbf{x}}$. In terms of normalization, note that $\left\|\underline{\beta} \circ \underline{\mathbf{x}}\right\| = \|\underline{\mathbf{x}}\|$ if $\mathtt{circ}(\underline{\beta})$ is an orthogonal circulant. This follows most easily by noting that

$$\left\|\underline{\beta} \circ \underline{\mathbf{x}}\right\| \leftrightarrow \left(\sum_{i=1}^{n} \mathtt{circ}(\underline{\beta})^* \, \mathtt{circ}(\mathbf{x}_i)^* \, \mathtt{circ}(\mathbf{x}_i) \, \mathtt{circ}(\underline{\beta}) \right)^{1/2} \leftrightarrow \|\underline{\mathbf{x}}\|,$$

because circulant matrices commute and $\mathtt{circ}(\underline{\beta})$ is orthogonal by construction. For this reason, we consider orthogonal circulant matrices the analogues of *angles* or *signs*, and normalized eigenvectors in the circulant algebra can be rescaled by them. (Recall that we showed that $\mathtt{angle}(\underline{\alpha})$ is an orthogonal circulant in Section 3.)

DOI: 10.1002/nla

The Fourier transform offers a convenient decoupling procedure to compute eigenvalues and eigenvectors, as observed by Braman [3]. Let $\underline{\boldsymbol{A}} \in \mathbb{K}_k^{n\times n}$ and let $\underline{\mathbf{x}} \in \mathbb{K}_k^n$ and $\underline{\lambda}$ be an eigenvalue and eigenvector pair: $\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}} = \underline{\mathbf{x}} \circ \underline{\lambda}$ and $\det(\underline{\boldsymbol{A}} - \underline{\lambda} \circ \underline{\boldsymbol{I}}) = 0$. Then it is straightforward to show that the Fourier transforms $\mathtt{cft}(\underline{\boldsymbol{A}})$, $\mathtt{cft}(\underline{\mathbf{x}})$, and $\mathtt{cft}(\underline{\lambda})$ decouple as follows:

$$\begin{aligned}
\mathtt{cft}(\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}}) &= \mathtt{cft}(\underline{\mathbf{x}} \circ \underline{\lambda});\\
\mathtt{cft}(\underline{\boldsymbol{A}})\,\mathtt{cft}(\underline{\mathbf{x}}) &= \mathtt{cft}(\underline{\mathbf{x}})\,\mathtt{cft}(\underline{\lambda});\\
\begin{bmatrix} \hat{\boldsymbol{A}}_1 & & \\ & \ddots & \\ & & \hat{\boldsymbol{A}}_k \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}_1 & & \\ & \ddots & \\ & & \hat{\mathbf{x}}_k \end{bmatrix} &= \begin{bmatrix} \hat{\mathbf{x}}_1 & & \\ & \ddots & \\ & & \hat{\mathbf{x}}_k \end{bmatrix} \begin{bmatrix} \hat{\lambda}_1 & & \\ & \ddots & \\ & & \hat{\lambda}_k \end{bmatrix};\\
\begin{bmatrix} \hat{\boldsymbol{A}}_1\hat{\mathbf{x}}_1 & & \\ & \ddots & \\ & & \hat{\boldsymbol{A}}_k\hat{\mathbf{x}}_k \end{bmatrix} &= \begin{bmatrix} \hat{\lambda}_1\hat{\mathbf{x}}_1 & & \\ & \ddots & \\ & & \hat{\lambda}_k\hat{\mathbf{x}}_k \end{bmatrix},
\end{aligned}$$

where $\hat{\lambda}_j \in \lambda(\hat{\boldsymbol{A}}_j)$ and $\hat{\mathbf{x}}_j \neq 0$. The last equation follows because

$$\mathtt{cft}(\det(\underline{\boldsymbol{A}} - \underline{\lambda} \circ \underline{\boldsymbol{I}})) = \operatorname{diag}\left[\, \det(\hat{\boldsymbol{A}}_1 - \hat{\lambda}_1 \boldsymbol{I}),\ \ldots,\ \det(\hat{\boldsymbol{A}}_k - \hat{\lambda}_k \boldsymbol{I}) \,\right] = 0.$$

The decoupling procedure we just described shows that *any* eigenvalue or eigenvector of $\underline{\boldsymbol{A}}$ must decompose into individual eigenvalues or eigenvectors of the `cft`-transformed problem. This illustrates a fundamental difference from the standard matrix algebra. For standard matrices, requiring $\det(\boldsymbol{A} - \lambda \boldsymbol{I}) = 0$ and finding a nonzero solution $\mathbf{x}$ for $\boldsymbol{A}\mathbf{x} = \lambda\mathbf{x}$ are equivalent. In contrast, the determinant and the eigenvector equations are not equivalent in the circulant algebra: $\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}} = \underline{\mathbf{x}} \circ \underline{\lambda}$ actually has an infinite number of solutions $\underline{\lambda}$. For instance, set $\hat{\mathbf{x}}_1, \hat{\lambda}_1$ to be an eigenpair of $\hat{\boldsymbol{A}}_1$ and $\hat{\mathbf{x}}_j = 0$ for $j > 1$, then any value for $\hat{\lambda}_j$ solves $\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}} = \underline{\mathbf{x}} \circ \underline{\lambda}$. However, only a few of these solutions also satisfy $\det(\underline{\boldsymbol{A}} - \underline{\lambda} \circ \underline{\boldsymbol{I}}) = \underline{0}$.

Eigenvalues of matrices in $\mathbb{K}_k^{n\times n}$ have some interesting properties. Most notably, a matrix may have more than n eigenvalues. As a special case, the diagonal elements of a matrix are not necessarily the only eigenvalues. We demonstrate these properties with an example.

Example 5
For the diagonal matrix

$$\begin{bmatrix} \{2\ 3\ 1\} & \{0\ 0\ 0\} \\ \{0\ 0\ 0\} & \{3\ 1\ 1\} \end{bmatrix}$$

we have

$$\hat{\boldsymbol{A}}_1 = \begin{bmatrix} 6 & 0 \\ 0 & 5 \end{bmatrix}, \quad \hat{\boldsymbol{A}}_2 = \begin{bmatrix} -\imath\sqrt{3} & 0 \\ 0 & 2 \end{bmatrix}, \quad \hat{\boldsymbol{A}}_3 = \begin{bmatrix} \imath\sqrt{3} & 0 \\ 0 & 2 \end{bmatrix}.$$

Thus,

$$\begin{aligned}
\underline{\lambda}_1 &= \mathtt{icft}(\operatorname{diag}\left[\,6\ 2\ 2\,\right]) = \ (1/3)\ \{10\ 4\ 4\} & \underline{\lambda}_2 &= \mathtt{icft}(\operatorname{diag}\left[\,5\ -\imath\sqrt{3}\ \imath\sqrt{3}\,\right]) = \ (1/3)\ \{5\ 2\ 2\}\\
\underline{\lambda}_3 &= \mathtt{icft}(\operatorname{diag}\left[\,6\ -\imath\sqrt{3}\ \imath\sqrt{3}\,\right]) = \{2\ 3\ 1\} & \underline{\lambda}_4 &= \mathtt{icft}(\operatorname{diag}\left[\,5\ 2\ 2\,\right]) = \ (1/3)\ \{3\ 1\ 1\}.
\end{aligned}$$

The corresponding eigenvectors are

$$\begin{aligned}
\underline{\mathbf{x}}_1 &= \begin{bmatrix} \{1/3\ 1/3\ 1/3\} \\ \{2/3\ -1/3\ -1/3\} \end{bmatrix}; & \underline{\mathbf{x}}_2 &= \begin{bmatrix} \{2/3\ -1/3\ -1/3\} \\ \{1/3\ 1/3\ 1/3\} \end{bmatrix};\\
\underline{\mathbf{x}}_3 &= \begin{bmatrix} \{1\ 0\ 0\} \\ \{0\ 0\ 0\} \end{bmatrix}; & \underline{\mathbf{x}}_4 &= \begin{bmatrix} \{0\ 0\ 0\} \\ \{1\ 0\ 0\} \end{bmatrix}.
\end{aligned}$$

There are still more eigenvalues, however. The four eigenvalues above all correspond to elements in $\mathbb{K}_k$ with real-valued entries. We can combine the eigenvalues of the $\hat{\boldsymbol{A}}_j$'s to produce complex-valued elements in $\mathbb{K}_k$ that are also eigenvalues. These are

$$\begin{aligned}
\underline{\lambda}_5 &= \mathtt{icft}(\operatorname{diag}\left[\,6\ -\imath\sqrt{3}\ 2\,\right]) & \underline{\lambda}_6 &= \mathtt{icft}(\operatorname{diag}\left[\,6\ 2\ \imath\sqrt{3}\,\right])\\
\underline{\lambda}_7 &= \mathtt{icft}(\operatorname{diag}\left[\,5\ -\imath\sqrt{3}\ 2\,\right]) & \underline{\lambda}_8 &= \mathtt{icft}(\operatorname{diag}\left[\,5\ 2\ \imath\sqrt{3}\,\right]).
\end{aligned}$$

DOI: 10.1002/nla

For completeness and further clarity, let us extend this example a bit by presenting also the eigenvalues of the non-diagonal matrix from Example 1. Let $\underline{\boldsymbol{A}} = \left[\begin{smallmatrix} \{2\ 3\ 1\} & \{8\ \text{-}2\ 0\} \\ \{\text{-}2\ 0\ 2\} & \{3\ 1\ 1\} \end{smallmatrix}\right]$. The `cft` produces:

$$\hat{\boldsymbol{A}}_1 = \begin{bmatrix} 6 & 6 \\ 0 & 5 \end{bmatrix}, \quad \hat{\boldsymbol{A}}_2 = \begin{bmatrix} \text{-}\sqrt{3} & \text{-}9 + \imath\sqrt{3} \\ \text{-}3 + \imath\sqrt{3} & 2 \end{bmatrix}, \quad \hat{\boldsymbol{A}}_3 = \begin{bmatrix} \imath\sqrt{3} & \text{-}9 - \imath\sqrt{3} \\ \text{-}3 + \imath\sqrt{3} & 2 \end{bmatrix}.$$

The numerical eigenvalues of $\hat{\boldsymbol{A}}_1$ are $\{6, 5\}$; of $\hat{\boldsymbol{A}}_2$ are $\{\text{-}0.0899 + 6.4282\imath, 2.0899 - 4.6962\imath\}$; and of $\hat{\boldsymbol{A}}_3$ are $\{\text{-}0.0899 - 6.4282\imath, 2.0899 + 4.6962\imath\}$. The real-valued eigenvalues of $\underline{\boldsymbol{A}}$ are

$$\underline{\lambda}_1 = \{1.9401\ \text{-}1.6814\ 5.7413\} \quad \underline{\lambda}_2 = \{3.0599\ 3.6814\ \text{-}1.7413\}$$
$$\underline{\lambda}_3 = \{3.3933\ 4.0147\ \text{-}1.4080\} \quad \underline{\lambda}_4 = \{1.6067\ \text{-}2.0147\ 5.4080\}.$$

The complex-valued eigenvalues of $\underline{\boldsymbol{A}}$ are

$$\underline{\lambda}_5 = \{4.6966 - 1.5654\imath,\ \text{-}0.7040 + 1.9114\imath,\ 2.0073 - 0.3461\imath\}$$
$$\underline{\lambda}_6 = \{3.6367 + 2.1427\imath,\ 3.0373 + 0.3980\imath,\ \text{-}0.6740 - 2.5407\imath\}$$
$$\underline{\lambda}_7 = \{4.3633 - 1.5654\imath,\ \text{-}1.0373 + 1.9114\imath,\ 1.6740 - 0.3461\imath\}$$
$$\underline{\lambda}_8 = \{3.3034 + 2.1427\imath,\ 2.7040 + 0.3980\imath,\ \text{-}1.0073 - 2.5407\imath\}.$$

We now count the number of unique eigenvalues and eigenvectors, using the decoupling procedure in the Fourier space. To simplify the discussion, let us only consider the case where each $\hat{\boldsymbol{A}}_j$ has simple eigenvalues. Consider an $\underline{\boldsymbol{A}} \in \mathbb{K}_k^{n\times n}$ with this property, and let m_j be the number of unique eigenvalues and eigenvectors of $\hat{\boldsymbol{A}}_j$. Then the number of unique eigenvalues of $\underline{\boldsymbol{A}}$ is given by the number of unique solutions to $\det(\underline{\boldsymbol{A}} - \underline{\lambda} \circ \underline{\boldsymbol{I}}) = 0$ which is $\prod_{j=1}^{k} m_j$. The number of unique eigenvectors (up to normalization) is given by the number of unique solutions to $\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}} = \underline{\mathbf{x}} \circ \underline{\lambda}$, which is also $\prod_{j=1}^{k} m_j$.

This result shows there are at most n^k eigenvalues if $\underline{\lambda} \in \mathbb{K}_k$ is allowed to be complex-valued, even when $\underline{\boldsymbol{A}} \in \mathbb{K}_k$ is real-valued. If $\underline{\boldsymbol{A}} \in \mathbb{K}_k$ is real-valued, then there are at most $n^{\lceil (k+1)/2 \rceil}$ "real" eigenvalues. For this result, note that $\texttt{icft}(\text{diag}\left[\begin{smallmatrix} \alpha_1 & \cdots & \alpha_k \end{smallmatrix}\right])$ is real-valued if and only if $\text{diag}\left[\begin{smallmatrix} \alpha_1 & \cdots & \alpha_k \end{smallmatrix}\right]^* = \boldsymbol{F}^2\, \text{diag}\left[\begin{smallmatrix} \alpha_1 & \cdots & \alpha_k \end{smallmatrix}\right] \boldsymbol{F}^2$ [32], where $\boldsymbol{F}$ is the Fourier transform matrix. This implies α_1 is real-valued, and $\alpha_j = \overline{\alpha_{k-j+1}}$. Applying this restriction reduces the feasible combinations of eigenvalues to $n^{\lceil (k+1)/2 \rceil}$.

Given that there are so many eigenvalues and vectors, are all of them necessary to describe $\underline{\boldsymbol{A}}$? We now show this is not the case by making a few definitions to clarify the discussion.

Definition 6
Let $\underline{\boldsymbol{A}} \in \mathbb{K}_k^{n\times n}$. A *canonical* set of eigenvalues and eigenvectors is a set of minimum size, ordered such that $\texttt{abs}(\underline{\lambda}_1) \geq \texttt{abs}(\underline{\lambda}_2) \geq \ldots \geq \texttt{abs}(\underline{\lambda}_k)$, which contains the information to reproduce *any* eigenvalue or eigenvector of $\underline{\boldsymbol{A}}$

In the diagonal matrix from Example 5, the sets $\{(\underline{\lambda}_1, \underline{\mathbf{x}}_1), (\underline{\lambda}_2, \underline{\mathbf{x}}_2)\}$, $\{(\underline{\lambda}_3, \underline{\mathbf{x}}_3), (\underline{\lambda}_4, \underline{\mathbf{x}}_4)\}$, and $\{(\underline{\lambda}_1, \underline{\mathbf{x}}_1), (\underline{\lambda}_3, \underline{\mathbf{x}}_3), (\underline{\lambda}_4, \underline{\mathbf{x}}_4)\}$ contain all the information to reproduce any eigenpair, whereas the set $\{(\underline{\lambda}_1, \underline{\mathbf{x}}_1), (\underline{\lambda}_3, \underline{\mathbf{x}}_3)\}$ does not (it does not contain the eigenvalue 5 of $\hat{\boldsymbol{A}}_1$). In this case, the only canonical set is $\{(\underline{\lambda}_1, \underline{\mathbf{x}}_1), (\underline{\lambda}_2, \underline{\mathbf{x}}_2)\}$. This occurs because, by a simple counting argument, a canonical set must have at least two eigenvalues, thus the set is of minimum size. The choice of $\underline{\lambda}_1$ and $\underline{\lambda}_2$ is given by the ordering condition. Among all the size 2 sets with all the information, this is the only one with the property that $\texttt{abs}(\underline{\lambda}_1) \geq \texttt{abs}(\underline{\lambda}_2)$.

Theorem 7 (Unique Canonical Decomposition)
Let $\underline{\boldsymbol{A}} \in \mathbb{K}_k^{n\times n}$ where each $\hat{\boldsymbol{A}}_j$ in the $\texttt{cft}(\underline{\boldsymbol{A}})$ matrix has distinct eigenvalues with distinct magnitudes. Then $\underline{\boldsymbol{A}}$ has a unique canonical set of n eigenvalues and eigenvectors. This canonical set corresponds to a basis of n eigenvectors, yielding an eigendecomposition

$$\underline{\boldsymbol{A}} = \underline{\boldsymbol{X}} \circ \underline{\boldsymbol{\Lambda}} \circ \underline{\boldsymbol{X}}^{-1}.$$

DOI: 10.1002/nla

Proof
Because all of the eigenvalues of each $\hat{\boldsymbol{A}}_j$ are distinct, with distinct magnitudes, there are nk distinct numbers. This implies that any canonical set must have at least n eigenvalues.

Let $\hat{\lambda}_j^{(i)}$ be the ith eigenvalue of $\hat{\boldsymbol{A}}_j$ ordered such that $|\hat{\lambda}_j^{(1)}| > |\hat{\lambda}_j^{(2)}| > \ldots > |\hat{\lambda}_j^{(n)}|$. Then $\underline{\lambda}_i = \texttt{icft}(\text{diag}\left[\hat{\lambda}_1^{(i)}, \quad \ldots, \quad \hat{\lambda}_k^{(i)}\right])$ is a canonical set of eigenvalues. We now show that this set constitutes an eigenbasis. Let $\hat{\boldsymbol{A}}_j = \hat{\boldsymbol{X}}_j \hat{\boldsymbol{\Lambda}}_j \hat{\boldsymbol{X}}_j^{-1}$ be the eigendecomposition using the magnitude ordering above. Then $\underline{\boldsymbol{X}} = \texttt{icft}(\text{diag}\left[\hat{\boldsymbol{X}}_1, \quad \ldots, \quad \hat{\boldsymbol{X}}_k\right])$ and $\underline{\boldsymbol{\Lambda}} = \texttt{icft}(\text{diag}\left[\hat{\boldsymbol{\Lambda}}_k, \quad \ldots, \quad \hat{\boldsymbol{\Lambda}}_k\right])$ is an eigenbasis because the matrix $\underline{\boldsymbol{X}}$ satisfies the properties of a basis from Theorem 3. Note that $\underline{\boldsymbol{\Lambda}}_{i,i} = \underline{\lambda}_i$.

Finally, we show that the set is unique. In any canonical set $\texttt{abs}(\underline{\lambda}_1) \geq \texttt{abs}(\underline{\lambda}_i)$ for $i > 1$. In the Fourier space, this implies $|\hat{\lambda}_j^{(1)}| \geq |\hat{\lambda}_j^{(i)}|$. Because all of the values $|\hat{\lambda}_j^{(i)}|$ are unique, there is no choice for $\hat{\lambda}_j^{(1)}$ in a canonical set and we have $|\hat{\lambda}_j^{(1)}| > |\hat{\lambda}_j^{(i)}|, i > 1$. Consequently, $\underline{\lambda}_1$ is unique. Repeating this argument on the remaining choices for $\underline{\lambda}_i$ shows that the entire set is unique. □

Remark 1
If $\hat{\boldsymbol{A}}_j$ has distinct eigenvalues but they do not have distinct magnitudes, then $\underline{\boldsymbol{A}}$ has an eigenbasis but the canonical set may not be unique, because $\hat{\boldsymbol{A}}_j$ may have two distinct eigenvalues with the same magnitude.

Next, we show that the eigendecomposition is *real-valued* under a surprisingly mild condition.

Theorem 8
Let $\underline{\boldsymbol{A}} \in \mathbb{K}_k^{n \times n}$ be real-valued with diagonalizable $\hat{\boldsymbol{A}}_j$ matrices. If k is odd, then the eigendecomposition $\underline{\boldsymbol{X}} \circ \underline{\boldsymbol{\Lambda}} \circ \underline{\boldsymbol{X}}^{-1}$ is real-valued if and only if $\hat{\boldsymbol{A}}_1$ has real-valued eigenvalues. If k is even, then $\underline{\boldsymbol{X}} \circ \underline{\boldsymbol{\Lambda}} \circ \underline{\boldsymbol{X}}^{-1}$ is real-valued if and only if $\hat{\boldsymbol{A}}_1$ and $\hat{\boldsymbol{A}}_{k/2+1}$ have real-valued eigenvalues.

Proof
First, if $\underline{\boldsymbol{A}}$ has a real-valued eigendecomposition, then we have that $\hat{\boldsymbol{X}}_1$ is real and also that $\hat{\boldsymbol{X}}_{k/2+1}$ is real when k is even. Likewise, $\hat{\boldsymbol{\Lambda}}_1$ is real and $\hat{\boldsymbol{\Lambda}}_{k/2+1}$ is real when k is even. Thus, $\hat{\boldsymbol{A}}_1$ (and also $\hat{\boldsymbol{A}}_{k/2+1}$ when k is even) have real-valued eigenvalues and vectors.

When $\hat{\boldsymbol{A}}_1$ (and $\hat{\boldsymbol{A}}_{k/2+1}$ for k even) have real-valued eigenvalues and vectors, then note that we can choose eigenvalues and eigenvectors of the other matrices $\hat{\boldsymbol{A}}_j$, which may be complex, in complex-conjugate pairs so as to satisfy the condition for a real-valued inverse Fourier transforms. This happens because when $\underline{\boldsymbol{A}}$ is real, then $\hat{\boldsymbol{A}}_1$ is real and $\hat{\boldsymbol{A}}_j = \overline{\hat{\boldsymbol{A}}_{k-j+2}}$ by the properties of the Fourier transform [32]. Thus for each eigenpair $\hat{\lambda}_j, \hat{\mathbf{x}}_j$ of $\hat{\boldsymbol{A}}_j$, the pair $\overline{\hat{\lambda}}_j, \overline{\hat{\mathbf{x}}}_j$ is an eigenpair for $\hat{\boldsymbol{A}}_{k-j+2}$. Consequently, if we always choose these complex conjugate pairs for all j besides $j = 1$ (and $j = k/2 + 1$ for k even), then the result of the inverse Fourier transform will be real-valued. □

Finally, we note that if the scalars of a matrix are padded with zeros to transform them into the circulant algebra, then the canonical set of eigenvalues are nothing but tuples that consist of the eigenvalues of the original matrix in the first entry, padded with zeros as well. To justify this observation, let $\underline{\boldsymbol{A}} \in \mathbb{K}_k^{n \times n}$ have $\underline{A}_{i,j} = \{G_{i,j}, 0, \ldots, 0\}$ for a matrix $\boldsymbol{G} \in \mathbb{R}^{n \times n}$. Also, let $\lambda_1, \ldots, \lambda_m \quad (m \leq n)$ be the eigenvalues of $\boldsymbol{G}$ ordered such that $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_m|$. Then $\texttt{cft}(\underline{A}_{i,j}) = \text{diag}\left[G_{i,j}, \ldots, G_{i,j}\right]$ and thus $\hat{\boldsymbol{A}}_j = \boldsymbol{G}$ for all j. Thus, we only need to combine the same m eigenvalues of each $\hat{\boldsymbol{A}}_j$ to construct eigenvalues of $\underline{\boldsymbol{A}}$. For the eigenvalues $\underline{\lambda}_i$, we have $\texttt{cft}(\underline{\lambda}) = \text{diag}\left[\lambda_i, \ldots, \lambda_i\right]$, thus the given set is canonical because of the same argument used in the proof of Theorem 7.

We end this section by noting that much of the above analysis can be generalized to non-simple eigenvalues and vectors using the Jordan canonical form of the $\hat{\boldsymbol{A}}_j$ matrices.

DOI: 10.1002/nla

Require: $\underline{\boldsymbol{A}}, \underline{\mathbf{x}}^{(0)}, \tau$

{Kept for alignment}

$\underline{\mathbf{x}}^{(0)} \leftarrow \underline{\mathbf{x}}^{(0)} \circ \left\|\underline{\mathbf{x}}^{(0)}\right\|^{-1}$

for $k = 1, \ldots$ until convergence **do**

  $\underline{\mathbf{y}}^{(k)} \leftarrow \underline{\boldsymbol{A}} \circ \underline{\mathbf{x}}^{(k-1)}$

  $\underline{\alpha}^{(k)} \leftarrow \left\|\underline{\mathbf{y}}^{(k)}\right\|$

  $\underline{\mathbf{x}}^{(k)} \leftarrow \underline{\mathbf{y}}^{(k)} \circ \underline{\alpha}^{(k)^{-1}}$

  if converged **then**

    return $\underline{\mathbf{x}}^{(k)}$

  end if

end for

Require: $\underline{\boldsymbol{A}}, \underline{\mathbf{x}}^{(0)}, \tau$

$\hat{\boldsymbol{A}} \leftarrow \mathtt{cft}(\underline{\boldsymbol{A}}), \hat{\boldsymbol{X}}^{(0)} \leftarrow \mathtt{cft}(\underline{\mathbf{x}}^{(0)})$

$\hat{\boldsymbol{X}}^{(0)} \leftarrow \hat{\boldsymbol{X}}^{(0)} \left(\hat{\boldsymbol{X}}^{(0)^*} \hat{\boldsymbol{X}}^{(0)}\right)^{-1/2}$

for $k = 1, \ldots$ until convergence **do**

  $\hat{\boldsymbol{Y}}^{(k)} \leftarrow \hat{\boldsymbol{A}} \hat{\boldsymbol{X}}^{(k-1)}$

  $\hat{\boldsymbol{R}}^{(k)} \leftarrow \hat{\boldsymbol{Y}}^{(k)^*} \hat{\boldsymbol{Y}}^{(k)}$

  $\hat{\boldsymbol{X}}^{(k)} \leftarrow \hat{\boldsymbol{Y}}^{(k)} \hat{\boldsymbol{R}}^{(k)^{-1/2}}$

  if converged **then**

    return $\mathtt{icft}(\hat{\boldsymbol{X}}^{(k)})$

  end if

end for

Figure 3. The power method in the circulant algebra (left) and the power method in the circulant algebra after transformation with the fast Fourier transform (right). We address convergence criteria in Section 5.1.

5. THE POWER METHOD AND THE ARNOLDI METHOD

In what follows, we show that the power method in the circulant algebra computes the eigenvalue $\underline{\lambda}_1$ in the canonical set of eigenvalues. This result shows how the circulant algebra matches the behavior of the standard power method. As part of our analysis, we show that the power method decouples into k independent power iterations in Fourier space and is equivalent to a subspace iteration method. Second, we demonstrate the Arnoldi method in the circulant algebra. In Fourier space, the Arnoldi method is also equivalent to the Arnoldi algorithm on independent problems, and it also corresponds to a particular block Arnoldi procedure.

5.1. The power method

Please see the left half of Figure 3 for the sequence of operations in the power method in the circulant algebra. In fact, it is not too different from the standard power method in Figure 1. We replace $\boldsymbol{A}\mathbf{x}$ with $\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}}$ and use the norm and inverse from Section 2. We'll return to the convergence criteria shortly. As we show next, the algorithm runs k independent power methods in Fourier space. Thus, the right half of Figure 3 shows the equivalent operations in Fourier space.

To analyze the power method, consider the key iterative operation in the power method when transformed into Fourier space:

$$\begin{aligned}
&\mathtt{cft}(\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}} \circ (\|\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}}\|)^{-1}) \\
&\quad = \mathtt{cft}(\underline{\boldsymbol{A}})\, \mathtt{cft}(\underline{\mathbf{x}})(\mathtt{cft}(\underline{\mathbf{x}})^*\, \mathtt{cft}(\underline{\mathbf{x}}))^{-1/2} \\
&\quad = \begin{bmatrix} \hat{\boldsymbol{A}}_1\hat{\mathbf{x}}_1 & & \\ & \ddots & \\ & & \hat{\boldsymbol{A}}_k\hat{\mathbf{x}}_k \end{bmatrix} \left(\begin{bmatrix} \hat{\boldsymbol{A}}_1\hat{\mathbf{x}}_1 & & \\ & \ddots & \\ & & \hat{\boldsymbol{A}}_k\hat{\mathbf{x}}_k \end{bmatrix}^* \begin{bmatrix} \hat{\boldsymbol{A}}_1\hat{\mathbf{x}}_1 & & \\ & \ddots & \\ & & \hat{\boldsymbol{A}}_k\hat{\mathbf{x}}_k \end{bmatrix} \right)^{-1/2}.
\end{aligned}$$

Now,

$$\begin{aligned}
\left(\begin{bmatrix} \hat{\boldsymbol{A}}_1\hat{\mathbf{x}}_1 & & \\ & \ddots & \\ & & \hat{\boldsymbol{A}}_k\hat{\mathbf{x}}_k \end{bmatrix}^* \begin{bmatrix} \hat{\boldsymbol{A}}_1\hat{\mathbf{x}}_1 & & \\ & \ddots & \\ & & \hat{\boldsymbol{A}}_k\hat{\mathbf{x}}_k \end{bmatrix} \right)^{-1/2} &= \begin{bmatrix} \hat{\mathbf{x}}_1^*\hat{\boldsymbol{A}}_1^*\hat{\boldsymbol{A}}_1\hat{\mathbf{x}}_1 & & \\ & \ddots & \\ & & \hat{\mathbf{x}}_k^*\hat{\boldsymbol{A}}_k^*\hat{\boldsymbol{A}}_k\hat{\mathbf{x}}_k \end{bmatrix}^{-1/2} \\
&= \begin{bmatrix} \|\hat{\boldsymbol{A}}_1\hat{\mathbf{x}}_1\|^{-1} & & \\ & \ddots & \\ & & \|\hat{\boldsymbol{A}}_k\hat{\mathbf{x}}_k\|^{-1} \end{bmatrix}.
\end{aligned}$$

Prepared using nlaauth.cls
DOI: 10.1002/nla

Thus

$$\texttt{cft}(\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}} \circ (\|\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}}\|)^{-1}) = \begin{bmatrix} \hat{\boldsymbol{A}}_1\hat{\mathbf{x}}_1/\|\hat{\boldsymbol{A}}_1\hat{\mathbf{x}}_1\| & & \\ & \ddots & \\ & & \hat{\boldsymbol{A}}_k\hat{\mathbf{x}}_k/\|\hat{\boldsymbol{A}}_1\hat{\mathbf{x}}_1\| \end{bmatrix}.$$

The key iterative operation, $\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}} \circ (\|\underline{\boldsymbol{A}} \circ \underline{\mathbf{x}}\|)^{-1}$, corresponds to *one step* of the standard power method on each matrix $\hat{\boldsymbol{A}}_j$. From this derivation, we arrive at the following theorem, whose proof follows immediately from the convergence proof of the power method for a matrix.

Theorem 9
Let $\underline{\boldsymbol{A}} \in \mathbb{K}_k^{n\times n}$ have a canonical set of eigenvalues $\underline{\lambda}_1, \ldots, \underline{\lambda}_n$ where $|\underline{\lambda}_1| > |\underline{\lambda}_2|$, then the power method in the circulant algebra convergences to an eigenvector $\underline{\mathbf{x}}_1$ with eigenvalue $\underline{\lambda}_1$.

A bit tangentially, an eigenpair in the Fourier space is a simple instance of a *multivariate* eigenvalue problem [35]. The general multivariate eigenvalue problem is

$$\sum_j \boldsymbol{A}_{i,j}\mathbf{x}_j = \lambda_i \mathbf{x}_i, \qquad i = 1, \ldots,$$

whereas we study the same system, albeit diagonal. Chu and Watterson [35] did study a power method for the more general problem and showed local convergence; however our diagonal situation is sufficiently simple for us to state stronger results.

Convergence criteria. A simple measure such as $\left\|\underline{\mathbf{x}}^{(k)} - \underline{\mathbf{x}}^{(k-1)}\right\| \leq \underline{\tau}$, with $\underline{\tau} = \{\tau\ 0\ \ldots\ 0\}$ will not detect convergence. As mentioned in the description of the standard power method in Figure 1, this test can fail when the eigenvector changes angle. Here, we have the more general notion of an angle for each element, and eigenvectors are unique up to a choice of angle. Thus, we first normalize angles before comparing the iterates. We use the convergence criteria

$$\left\|\texttt{angle}(\underline{\mathbf{x}}_1^{(k)})^{-1} \circ \underline{\mathbf{x}}^{(k)} - \texttt{angle}(\underline{\mathbf{x}}_1^{(k-1)})^{-1} \circ \underline{\mathbf{x}}^{(k-1)}\right\| < \underline{\tau}. \tag{6}$$

In the Fourier space, this choice requires that *all* of the independent problems have converged to a tolerance of τ, which is a viable practical choice. An alternative convergence criteria is to terminate when the *eigenvalue* stops changing, although this may occur significantly before the eigenvector has converged.

Subspace iteration. We now show that the power method is equivalent to subspace iteration in Fourier space. Subspace iteration is also known as "orthogonal iteration" or the "block-power method." Given a starting block of vectors $\boldsymbol{X}^{(0)}$, the iteration is

$$\boldsymbol{Y} \leftarrow \boldsymbol{A}\boldsymbol{X}^{(k)}, \qquad \boldsymbol{X}^{(k+1)}, \boldsymbol{R}^{(k+1)} = \texttt{qr}(\boldsymbol{Y}).$$

On the surface, there is nothing to relate this iteration to our power method, even in Fourier space. The relationship, however, follows because all of our operations in Fourier space occur with *block-diagonal* matrices. Note that for a block-diagonal matrix of vectors, which is what $\hat{\boldsymbol{X}}^{(k)}$ is, the QR factorization just normalizes each column. In other words, the result is a diagonal matrix $\boldsymbol{R}$. This simplification shows that steps 5-6 in the Fourier space algorithm are equivalent to the QR factorization in subspace iteration.

Breakdown. One problem with this iterative approach is that it can encounter "zero divisors" as scalars when running these algorithms. These occur when the matrices in Fourier space are not invertible. We have not explicitly addressed this situation and note that the same issues arise in block methods when some of the quantities become singular. The analogy with the block method may provide an appropriate solution. For example, if the scalar $\underline{\alpha}^{(k)}$ is a zero-divisor, then we could use the QR factorization of $\hat{\boldsymbol{Y}}^{(k)}$ – as suggested by the equivalence with subspace iteration – instead.

Prepared using nlaauth.cls
DOI: 10.1002/nla

(a) Arnoldi for $\mathbb{R}^{n\times n}$

Require: $\boldsymbol{A}, \mathbf{b}, t$

$\mathbf{q}_1 \leftarrow \mathbf{b}/\|\mathbf{b}\|$
for $j = 1, \ldots, t$ **do**
  $\mathbf{z} \leftarrow \boldsymbol{A}\mathbf{q}_1$
  for $i = 1, \ldots, i$ **do**
    $H_{i,j} \leftarrow \mathbf{q}_i^* \mathbf{z}$
    $\mathbf{z} \leftarrow \mathbf{z} - H_{i,j}\mathbf{q}_i$
  end for
  $H_{j+1,j} \leftarrow \|\mathbf{z}\|$
  $\mathbf{q}_{j+1} \leftarrow \mathbf{z}/H_{j+1,j}$
end for

(b) Arnoldi for $\mathbb{K}_k^{n\times n}$

Require: $\underline{\boldsymbol{A}}, \underline{\mathbf{b}}, t$

$\underline{\mathbf{q}}_1 \leftarrow \mathbf{b} \circ \|\underline{\mathbf{b}}\|^{-1}$
for $j = 1, \ldots, t$ **do**
  $\underline{\mathbf{z}} \leftarrow \underline{\boldsymbol{A}} \circ \underline{\mathbf{q}}_j$
  for $i = 1, \ldots, j$ **do**
    $\underline{H}_{i,j} \leftarrow \langle \underline{\mathbf{q}}_i, \underline{\mathbf{z}} \rangle$
    $\underline{\mathbf{z}} \leftarrow \underline{\mathbf{z}} - \underline{H}_{i,j} \circ \underline{\mathbf{q}}_i$
  end for
  $\underline{H}_{j+1,j} \leftarrow \|\underline{\mathbf{z}}\|$
  $\underline{\mathbf{q}}_{j+1} \leftarrow \underline{\mathbf{z}} \circ \underline{H}_{j+1,j}^{-1}$
end for

(c) Unrolled Arnoldi for $\mathbb{K}_k^{n\times n}$

Require: $\underline{\boldsymbol{A}}, \underline{\mathbf{b}}, t$

$\hat{\boldsymbol{A}} \leftarrow \mathtt{cft}(\underline{\boldsymbol{A}})$
$\hat{\boldsymbol{B}} \leftarrow \mathtt{cft}(\underline{\mathbf{b}})$
$\hat{\boldsymbol{Q}}_1 \leftarrow \hat{\boldsymbol{B}}(\hat{\boldsymbol{B}}^* \hat{\boldsymbol{B}})^{-1/2}$
for $j = 1, \ldots, t$ **do**
  $\hat{\boldsymbol{Z}} \leftarrow \hat{\boldsymbol{A}}\hat{\boldsymbol{Q}}_j$
  for $i = 1, \ldots, j$ **do**
    $\hat{\boldsymbol{H}}_{i,j} \leftarrow \hat{\boldsymbol{Q}}_i^* \hat{\boldsymbol{Z}}$
    $\underline{\boldsymbol{H}}_{i,j} \leftarrow \mathtt{icft}(\hat{\boldsymbol{H}}_{i,j})$
    $\hat{\boldsymbol{Z}} \leftarrow \hat{\boldsymbol{Z}} - \hat{\boldsymbol{Q}}_i \hat{\boldsymbol{H}}_{i,j}$
  end for
  $\hat{\boldsymbol{H}}_{j+1,j} \leftarrow (\hat{\boldsymbol{Z}}^* \hat{\boldsymbol{Z}})^{1/2}$
  $\underline{\boldsymbol{H}}_{j+1,j} \leftarrow \mathtt{icft}(\hat{\boldsymbol{H}}_{j+1,j})$
  $\hat{\boldsymbol{Q}}_{j+1} \leftarrow \hat{\boldsymbol{Z}} \hat{\boldsymbol{H}}_{j+1,j}^{-1}$
  $\underline{\mathbf{q}}_{j+1} \leftarrow \mathtt{icft}(\hat{\boldsymbol{Q}}_{j+1})$
end for

Figure 4. Arnoldi methods. Algorithm (a) shows the standard Arnoldi process. Algorithm (b) shows the Arnoldi process in the circulant algebra, and Algorithm (c) shows the set of operations in (b) but expressed in the Fourier space.

5.2. The Arnoldi process

The Arnoldi method is a cornerstone of modern matrix computations. Let $\boldsymbol{A}$ be an $n \times n$ matrix with real valued entries. Then the Arnoldi method is a technique to build an orthogonal basis for the Krylov subspace

$$\mathcal{K}_t(\boldsymbol{A}, \mathbf{v}) = \mathtt{span}\{\mathbf{v}, \boldsymbol{A}\mathbf{v}, \ldots, \boldsymbol{A}^{t-1}\mathbf{v}\},$$

where $\mathbf{v}$ is an initial vector. Instead of using this power basis, the Arnoldi process derives a set of orthogonal vectors that span the same space when computed with exact arithmetic. The standard method is presented in Figure 4(a). From this procedure, we have the Arnoldi decomposition of a matrix:

$$\boldsymbol{A}\boldsymbol{Q}_t = \boldsymbol{Q}_{t+1}\boldsymbol{H}_{t+1,t}$$

where $\boldsymbol{Q}_t$ is an $n \times t$ matrix, and $\boldsymbol{H}_{t+1,t}$ is a $(t+1) \times t$ upper Hessenberg matrix. Arnoldi's orthogonal subspaces $\boldsymbol{Q}$ enable efficient algorithms for both solving large scale linear systems [5] and computing eigenvalues and eigenvectors [7].

Using our repertoire of operations, the Arnoldi method in the circulant algebra is presented in Figure 4(b). The circulant Arnoldi process decoupled via the `cft` is also shown in Figure 4(c).

We make three observations here. First, the decoupled (`cft`) circulant Arnoldi process is equivalent to individual Arnoldi processes on each matrix $\hat{\boldsymbol{A}}_j$. This follows by a similar analysis used to show the decoupling result about the power method. The verification of this fact for the Arnoldi iteration is a bit more tedious and thus we omit this analysis. Second, the same decoupled process is equivalent to a block Arnoldi process. This also follows for the same reason the equivalent result held for the power method: the QR factorization of a block-diagonal matrix-of-vectors is just a normalization of each vector. Third, we produce an Arnoldi factorization:

$$\underline{\boldsymbol{A}} \circ \underline{\boldsymbol{Q}}_t = \underline{\boldsymbol{Q}}_{t+1} \circ \underline{\boldsymbol{H}}_{t+1,t}.$$

In fact, this outcome is a corollary of the first property and follows from applying `icft` to the same analysis.

This discussion raises an interesting question, why iterate on all problems simultaneously? One case where this is advantageous is with sparse problems; and we return to this issue in the concluding discussion (Section 8).

DOI: 10.1002/nla

6. A MATLAB PACKAGE

The MATLAB environment is a convenient playground for algorithms involving matrices. We have extended it with a new class implementing the circulant algebra as a native MATLAB object. The name of the resulting package and class is `camat`: *circulant algebra matrix*. While we will show some non-trivial examples of our package later, let us start with a small example to give the flavor of how it works.

```
A = cazeros(2,2,3); % creates a camat type
A(1,1) = cascalar([2,3,1]); A(1,2) = cascalar([8,-2,0]);
A(2,1) = cascalar([-2,0,2]); A(2,2) = cascalar([3,1,1]);
eig(A)          % compute eigenvalues as in Example 2;
```

The output, which matches the non-diagonal matrix in Example 5, is:

```
ans =
(:,:,1) =  % the first eigenvalue
    1.9401
   -1.6814
    5.7413
(:,:,2) =  % the second eigenvalue
    3.0599
    3.6814
   -1.7413
```

Internally, each element $\underline{\boldsymbol{A}} \in \mathbb{K}_k^{m\times n}$ is stored as a $k \times n \times m$ array *along with* its `cft` transformed data. Each scalar is stored by the k parameters defining it. To describe this storage, let us introduce the notation

$$\texttt{vec}(\underline{\alpha}) \equiv \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{bmatrix} = \texttt{circ}(\underline{\alpha})\mathbf{e}_1,$$

to label the vector of k parameters explicitly. Thus, we store $\texttt{vec}(\underline{\alpha})$ for $\alpha \in \mathbb{K}_k$. This storage corresponds to storing each scalar $\mathbb{K}_k$ consecutively in memory. The matrix is then stored by rows. We store the data for the diagonal elements of the `cft` transformed version in the same manner; that is, $\text{diag}(\texttt{cft}(\underline{\alpha}))$ is stored as k consecutive complex-valued scalars. The organization of matrices and vectors for the `cft` data is also *by row*. The reason we store the data by row is so we can take advantage of MATLAB's standard display operations.

At the moment, our implementation stores the elements in both the standard and Fourier transformed space. The rationale behind this choice was to make it easy to investigate the results in this manuscript. Due to the simplicity of the operations in the Fourier space, most of the functions on `camat` objections use the Fourier coefficients to compute a result efficiently and then compute the inverse Fourier transform for the `vec` representation. Hence, rather than incurring for the Fourier transform and inverse Fourier transform cost for each operation, we only incur the cost of the inverse transform. Because so few operations are easier in the standard space, we hope to eliminate the standard `vec` storage in a future version of the code to accelerate it even further.

We now show how the overloaded operation `eig` works in Figure 5. This procedure, inspired by Theorem 8, implements the process to get real-valued canonical eigenvalues and eigenvectors of a real-valued matrix in the circulant algebra. The slice `Af(j,:,:)` is the matrix $\hat{\boldsymbol{A}}_j^T$. Here, the *real-valued* transpose results from the storage-by-rows instead of the storage-by-columns. The code proceeds by computing the eigendecomposition of each $\hat{\boldsymbol{A}}_j$ with a special sort applied to produce the canonical eigenvalues. After all of the eigendecompositions are finished, we need to transpose their output. Then it feeds them to the `ifft` function to generate the data in `vec` form.

DOI: 10.1002/nla

```
function [V,D] = eig(A)
% CAEIG The eigenvalue routine in the circulant algebra
Af = A.fft; k = size(Af,1);           % extract data from object
if any(imag(A.data(:))), error('specialized for real values'); end
[Vf,Df] = deal(zeros(Af));            % allocate data of size (k,n,n)
[Vf(1,:,:),Df(1,:,:)] = sortedeig(squeeze(Af(1,:,:)).');
for j=2:floor(k/2)+1
  [Vf(j,:,:),Df(j,:,:)] = sortedeig(sqeeze(Af(j,:,:)).');
  if j~=k/2+1                         % skip last when k is even
    Vf(k-j+2,:,:) = conj(Vf(j,:,:)); Df(k-j+2,:,:) = conj(Df(j,:,:));
  end
end
% transpose all the data back.
for j=1:k, Vf(j,:,:) = Vf(j,:,:).'; Df(j,:,:) = Df(j,:,:).'; end
V = camatcft(ifft(Vf),Vf);            % create classed output
D = camatcft(ifft(Df),Df);
function [V,D]=sortedeig(A)
[V,D] = eig(A); d = diag(D); [ignore p] = sort(-abs(d));
V = V(:,p); D = D(p,p);               % apply the sort
```

Figure 5. The implementation of the eigenvalue computation in our package.

```
for iter=1:maxiter
  Ax = A*x;
  lambda = x'*Ax;
  x2 = (1./ norm(Ax))*Ax;
  delta = mag(norm(1./angle(x(1))*x-1./angle(x2(1))*x2));
  if delta<tol, break, end
end
```

Figure 6. The implementation of the power method using our package.

In a similar manner, we overloaded the standard assignment and indexing operations e.g. `a = A(i,j); A(1,1) = a`; the standard Matlab arithmetic operations `+, -, *, /, \`; and the functions `abs, angle, norm, conj, diag, eig, hess, mag, norm, numel, qr, rank, size, sqrt, svd`.

All of these operations have been mentioned or are self explanatory, except `mag`. It is a *magnitude function*, and we discuss it in detail in A.

Using these overloaded operations, implementing the power method is straightforward; see Figure 6. We note that the power method and Arnoldi methods can be further optimized by implementing them directly in Fourier space. This remains as an item for future work.

7. NUMERICAL EXAMPLES

In this section, we present a numerical example using the code we described in Section 6. The problem we consider is the Poisson equation on a regular grid with a mixture of periodic and fixed boundary conditions:

$$-\Delta u(x,y) = f(x,y) \qquad u(x,0) = u(x,1), u(0,y) = y(1,y) = 0 \qquad (x,y) \in [0,1] \times [0,1].$$

Consider a uniform mesh and the standard 5-point discrete Laplacian:

$$-\Delta u(x_i,y_j) \approx -u(x_{i-1},y_j) - u(x_i,y_{j-1}) + 4u(x_i,y_j) - u(x_{i+1},y_j) - u(x_i,y_{j+1}).$$

Prepared using nlaauth.cls
DOI: 10.1002/nla

After applying the boundary conditions and organizing the unknowns of u in y-major order, an approximate solution u is given by solving an $N(N-1) \times N(N-1)$ block-tridiagonal, circulant-block system:

$$\underbrace{\begin{bmatrix} \boldsymbol{C} & -\boldsymbol{I} & & \\ -\boldsymbol{I} & \boldsymbol{C} & \ddots & \\ & \ddots & \ddots & -\boldsymbol{I} \\ & & -\boldsymbol{I} & \boldsymbol{C} \end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix} \mathbf{u}(x_1,\cdot) \\ \mathbf{u}(x_2,\cdot) \\ \vdots \\ \mathbf{u}(x_{N-1},\cdot) \end{bmatrix}}_{\mathbf{u}} = \underbrace{\begin{bmatrix} \mathbf{f}(x_1,\cdot) \\ \mathbf{f}(x_2,\cdot) \\ \vdots \\ \mathbf{f}(x_{N-1},\cdot) \end{bmatrix}}_{\mathbf{f}}, \qquad \boldsymbol{C} = \underbrace{\begin{bmatrix} 4 & -1 & & -1 \\ -1 & 4 & \ddots & \\ & \ddots & \ddots & -1 \\ -1 & & -1 & 4 \end{bmatrix}}_{N \times N},$$

that is, $\boldsymbol{A}\mathbf{u} = \mathbf{f}$. Because of the circulant-block structure, this system is equivalent to

$$\underline{\boldsymbol{A}} \circ \underline{\mathbf{u}} = \underline{\mathbf{f}}$$

where $\underline{\boldsymbol{A}}$ is an $N-1 \times N-1$ matrix of $\mathbb{K}_N$ elements, $\underline{\mathbf{u}}$ and $\underline{\mathbf{f}}$ have compatible sizes, and

$$\boldsymbol{A} = \mathtt{circ}(\underline{\boldsymbol{A}}) \qquad \mathbf{u} = \mathtt{vec}(\underline{\mathbf{u}}) \qquad \mathbf{f} = \mathtt{vec}(\underline{\mathbf{f}}).$$

We now investigate this matrix and linear system with $N = 50$.

We acknowledge that the problem in this form is easy to solve by other means, for example by fast Poisson solvers based on the Fourier decompositions, or multi-grid methods. Our claim here is not that the approach we take is faster; rather, it is an illustration of the theory from the previous section. For this reason, we use a problem where the eigenvalues and eigenvectors are explicitly available. Others have also considered a similar example [30, 24].

7.1. The power method

We first study the behavior of the power method on $\underline{\boldsymbol{A}}$. The canonical eigenvalues of $\underline{\boldsymbol{A}}$ are

$$\underline{\lambda}_j = \left\{ 4+2\cos(j\pi/N), -1, 0, \ldots, 0, -1 \right\}.$$

To see this result, let $\underline{\lambda}(\mu) = \left\{ \mu, -1, 0, \ldots, 0, -1 \right\}$. Then

$$(\underline{\boldsymbol{A}} - \underline{\lambda}(\mu) \circ \underline{\boldsymbol{I}}) = \begin{bmatrix} (4-\mu)\circ\underline{1} & -1\circ\underline{1} & & \\ -1\circ\underline{1} & (4-\mu)\circ\underline{1} & \ddots & \\ & \ddots & \ddots & -1\circ\underline{1} \\ & & -1\circ\underline{1} & (4-\mu)\circ\underline{1} \end{bmatrix}.$$

The *canonical* eigenvalues of $\underline{\boldsymbol{A}} - \underline{\lambda}(\mu) \circ \underline{\boldsymbol{I}}$ can be determined by choosing μ to be an eigenvalue of $\boldsymbol{T} = \mathtt{tridiag}(-1, 4, -1)$. These are given by setting $\mu = 4 + 2\cos(j\pi/N)$, where each choice $j = 1, \ldots, N-1$ produces a canonical eigenvalue $\underline{\lambda}_j$. From these canonical eigenvalues, we can estimate the convergence behavior of the power method. Recall that the algorithm runs independent power methods in Fourier space. Consequently, these rates are given by $\hat{\lambda}_2/\hat{\lambda}_1$ for each matrix $\hat{\boldsymbol{A}}_j$. To state these ratios compactly, let $\gamma_1 = 4 + 2\cos(\pi/N)$ and $\gamma_2 = 4 + 2\cos(2\pi/N)$; also let $\delta_j = 2\cos(-\pi + 2\pi(j-1)/N)$. For N even,

$$\begin{aligned} \mathtt{cft}(\underline{\lambda}_1) &= \mathrm{diag}\left[\gamma_1+\delta_1, \ldots, \gamma_1+\delta_N \right] \\ \mathtt{cft}(\underline{\lambda}_2) &= \mathrm{diag}\left[\gamma_2+\delta_1, \ldots, \gamma_2+\delta_N \right] \end{aligned}$$

Thus, the convergence ratio for $\hat{\boldsymbol{A}}_j$ is $(\gamma_2 + \delta_j)/(\gamma_1 + \delta_j)$. The largest ratio (fastest converging) corresponds to the smallest value of δ_j, which is δ_1. The smallest ratio (slowest converging) corresponds to the largest value of δ_j, which is $\delta_{N/2+1}$ in this case. (This choice will slightly

DOI: 10.1002/nla

change in an obvious manner if N is odd.) Evaluating these ratios yields

$$\min_j \frac{\lambda_2(\hat{\boldsymbol{A}}_j)}{\lambda_1(\hat{\boldsymbol{A}}_j)} = \frac{\gamma_2+\delta_1}{\gamma_1+\delta_1} = \frac{2+2\cos(2\pi/N)}{2+2\cos(\pi/N)} \qquad \text{(fastest)}$$

$$\max_j \frac{\lambda_2(\hat{\boldsymbol{A}}_j)}{\lambda_1(\hat{\boldsymbol{A}}_j)} = \frac{\gamma_2+\delta_{N/2+1}}{\gamma_1+\delta_{N/2+1}} = \frac{6+2\cos(2\pi/N)}{6+2\cos(\pi/N)} \qquad \text{(slowest)}.$$

Based on this analysis, we expect the eigenvector to converge linearly with the rate $\frac{6+2\cos(2\pi/N)}{6+2\cos(\pi/N)}$. By the standard theory for the power method, expect the eigenvalues to converge twice as fast.

Let $\underline{\rho}$ be the eigenvector change measure from equation (6). In Figure 7, we first show how the maximum absolute value of the Fourier coefficients in $\underline{\rho}$ behaves (the red line). Formally, this measure is $\left\|\mathtt{cft}(\underline{\rho})\right\|_1$, i.e., the maximum element in the diagonal matrix. We also show how each Fourier component of the eigenvalue converges to the Fourier components of $\underline{\lambda}_1$ (each gray line). That is, let $\underline{\mu}^{(i)}$ be the Rayleigh quotient $\underline{\mathbf{x}}^{(i)^*} \circ \underline{\boldsymbol{A}} \circ \underline{\mathbf{x}}^{(i)}$ at the ith iteration. Then these lines are the N values of $\mathrm{diag}(\mathtt{cft}(\mathtt{abs}(\underline{\mu}^{(i)} - \underline{\lambda}_1)))$. The results validate the theoretical predictions, and the eigenvalue does indeed converge to $\underline{\lambda}_1$.

7.2. The Arnoldi method

We next investigate computing $\underline{\mathbf{u}}$ using the Arnoldi method applied to $\underline{\boldsymbol{A}}$. In this case, $f(x,y)$ to be 1 at x_{25}, y_2 and 0 elsewhere. This corresponds to a single non-zero in $\mathtt{vec}(\underline{\mathbf{f}})$ with value $1/N^2$. With this right-hand side, the procedure we use is identical to an unoptimized GMRES procedure. Given a t-step Arnoldi factorization starting from $\underline{\mathbf{f}}$, we estimate

$$\underline{\mathbf{u}}^{(t)} \approx \underline{\boldsymbol{Q}}_t \circ \arg\min_{\underline{\mathbf{y}} \in \mathbb{K}_k} \left\| \underline{\boldsymbol{H}}_{t+1,t} \circ \underline{\mathbf{y}} - \underline{\beta} \circ \underline{\mathbf{e}}_1 \right\|,$$

where $\underline{\beta} = \|\underline{\mathbf{f}}\|$. We solve the least-squares problem by solving each problem independently in the Fourier space – as has become standard throughout this paper. Let $\underline{\rho} = \left\|\underline{\mathbf{f}} - \underline{\boldsymbol{A}} \circ \underline{\mathbf{u}}^{(t)}\right\|$. Figure 8 shows (in red) the magnitude of the residual as a function of the Arnoldi factorization length t, which is $\left\|\mathtt{cft}(\underline{\rho})\right\|_1$. The figure also shows (in gray) the magnitude of the error in the jth Fourier coefficient; these lines are the N values of $\mathrm{diag}(\mathtt{cft}(\left\|\underline{\mathbf{u}} - \underline{\mathbf{u}}^{(t)}\right\|))$. In Fourier space, these values measure the error in each individual Arnoldi process.

What the figure shows is that the residual suddenly converges at the 26th iteration. This is in fact theoretically expected [36], because each matrix $\hat{\boldsymbol{A}}_j$ has $N/2+1 = 26$ distinct eigenvalues. In terms of measure the individual errors (the gray lines), some converge rapidly, and some do not seem to converge at all until the Arnoldi process completes at iteration 26. This exemplifies how the overall behavior is governed by the worst behavior in any of the independent Arnoldi processes.

8. CONCLUSION AND FUTURE WORK

We have extended the circulant algebra, introduced by Kilmer et al. [1], with new operations to pave the way for iterative algorithms, such as the power method and the Arnoldi iteration that we introduced. These operations provided key tools to build a MATLAB package to investigate these iterative algorithms for this paper. Furthermore, we used the fast Fourier transform to accelerate these operations, and as a key analysis tool for eigenvalues and eigenvectors. In the Fourier space the operations and algorithms decouple into individual problems. We observed this for the inner product, eigenvalues, eigenvectors, the power method, and the Arnoldi iteration. We also found that this decoupling explained the behavior in a numerical example.

DOI: 10.1002/nla

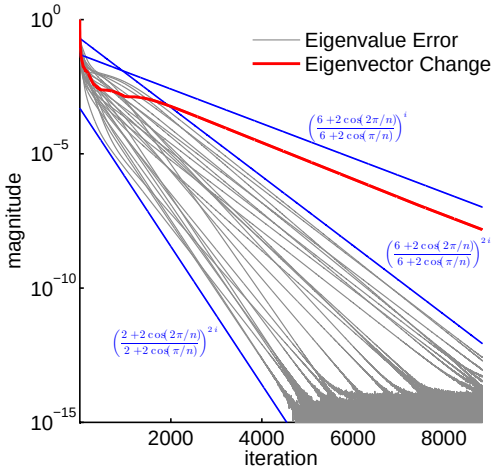

Figure 7. The convergence behavior of the power method in the circulant algebra. The gray lines show the error in the each eigenvalue in Fourier space. These curves track the predictions made based on the eigenvalues as discussed in the text. The red line shows the magnitude of the change in the eigenvector. We use this as the stopping criteria. It also decays as predicted by the ratio of eigenvalues. The blue fit lines have been visually adjusted to match the behavior in the convergence tail.

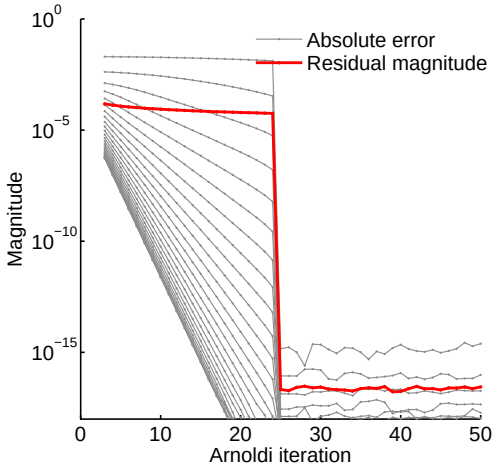

Figure 8. The convergence behavior of a GMRES procedure using the circulant Arnoldi process. The gray lines show the error in each Fourier component and the red line shows the magnitude of the residual. We observe poor convergence in one Fourier component; until the Arnoldi basis captures all of the eigenvalues after $N/2+1 = 26$ iterations. These results show how the two computations are performing individual power methods or Arnoldi processes in Fourier space.

Given that decoupling is such a powerful computational and analytical tool, a natural question that arises is when it is useful to employ the original circulant formalism, rather than work in the Fourier space. For dense computations, it is likely that working entirely in Fourier space is a superior approach. However, for sparse computations, such as the system $\underline{\boldsymbol{A}} \circ \underline{\mathbf{u}} = \underline{\mathbf{f}}$ explored in Section 7, such a conclusion is unwarranted. That example is sparse both in the matrix over circulants, and in the individual circulant arrays. When thought of as a cube of data, it is sparse in any way of slicing it into a matrix. After this matrix $\underline{\boldsymbol{A}}$ is transformed to the Fourier space, it loses its sparsity in the third-dimension; each sparse scalar $\underline{A}_{i,j}$ becomes a dense array. In this case, retaining the coupled nature of the operations and even avoiding most of the Fourier domain may allow better scalability in terms of total memory usage.

An interesting topic for future work is exploring other rings besides the ring of circulants, or even more general transformations from a tensor to a matrix as discussed by Brazell [24]. One obvious candidate is the ring of symmetric circulant matrices. In this ring, the Fourier coefficients are always real-valued. Using this ring avoids the algebraic and computational complexity associated with complex values in the Fourier transforms.

We have made all of the code and experiments available to use and reproduce our results:

`http://www.cs.purdue.edu/homes/dgleich/codes/camat`

REFERENCES

1. Kilmer ME, Martin CD, Perrone L. A third-order generalization of the matrix SVD as a product of third-order tensors. *Technical Report TR-2008-4*, Tufts University 2008. URL `http://www.cs.tufts.edu/tech_reports/reports/2008-4/report.pdf`.
2. Kilmer ME, Martin CD. Factorization strategies for third-order tensors. *Linear Algebra and its Applications* 2011; **435**(3):641–658, doi:10.1016/j.laa.2010.09.020.

DOI: 10.1002/nla

3. Braman K. Third-order tensors as linear operators on a space of matrices. *Linear Algebra and its Applications* 2010; **433**(7):1241–1253, doi:10.1016/j.laa.2010.05.025. URL `http://www.mcs.sdsmt.edu/kbraman/Research/Teigs2-submitted.pdf`.
4. von Mises R, Pollaczek-Geiringer H. Praktische verfahren der gleichungsauflösung. *Zeitschrift für Angewandte Mathematik und Mechanik* 1929; **9**(2):152–164, doi:10.1002/zamm.19290090206.
5. Krylov AN. On the numerical solution of the equation by which in technical questions frequencies of small oscillations of material systems are determined (Russian). *Izv. Akad. Nauk SSSR* 1931; **VII**(4):491–539.
6. Lanczos C. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *Journal of Research of the National Bureau of Standards* October 1950; **45**(4):255–282. URL `http://nvl.nist.gov/pub/nistpubs/jres/045/4/V45.N04.A01.pdf`.
7. Arnoldi WE. The principle of minimized iterations in the solution of the matrix eigenvalue problem. *Quarterly of Applied Mathematics* 1951; **9**:17–29.
8. Kilmer ME, Braman K, Hao N. Third order tensors as operators on matrices: A theoretical and computational framework with applications in imaging. *Technical Report CS-TR-01-11*, Tufts University 2011. URL `http://www.cs.tufts.edu/tech_reports/reports/2011-01/report.pdf`.
9. Lambert RH. Multichannel blind deconvolution: FIR matrix algebra and separation of multipath mixtures. PhD Thesis, University of Southern California May 1996.
10. Lambert RH, Joho M, Mathis H. Polynomial singular values for number of wideband sources estimation and principal component analysis'. *International Conference on Independent Component Analysis and Blind Signal Separation ICA*, 2001; 379–383. URL `http://ica-bss.org/joho/research/publications/ica_01.pdf`.
11. Mcdonald BR. *Linear algebra over commutative rings*. No. 87 in Pure and applied mathematics, Marcel Dekker, 1984.
12. Brewer JW, Bunce JW, Vleck FS. *Linear systems over commutative rings*, *Lecture notes in pure and applied mathematics*, vol. 104. Marcel Dekker, 1986. URL `http://books.google.com/books?id=xddbW1kj4zYC`.
13. Gustafson WH. Roth's theorems over commutative rings. *Linear Algebra and its Applications* February 1979; **23**:245–251, doi:10.1016/0024-3795(79)90106-X. URL `http://www.sciencedirect.com/science/article/B6V0R-45FKGH1-9R/2/c5ca43747018e55b16f944f31a4d014e`.
14. Prasad KM. Generalized inverses of matrices over commutative rings. *Linear Algebra and its Applications* November 1994; **211**:35–52, doi:10.1016/0024-3795(94)90081-7. URL `http://www.sciencedirect.com/science/article/B6V0R-45DHWPS-H/2/90bdae2d4c656d3bc1a2fa23478193e1`.
15. Gurvits L, Rodman L, Shalom T. Controllability and completion of partial upper triangular matrices over rings. *Linear Algebra and its Applications* July 1992; **172**:135–149, doi:10.1016/0024-3795(92)90023-4. URL `http://www.sciencedirect.com/science/article/B6V0R-45F5VYC-CS/2/25906e8df1f7a763fd1c37b3f900f8e9`.
16. Hafner JL, McCurley KS. Asymptotically fast triangularization of matrices over rings. *SIAM J. Comput.* December 1991; **20**(6):1068–1083, doi:10.1137/0220067.
17. Sontag ED. Linear systems over commutative rings: A survey. *Ricerche di Automatica* 1976; **1**:1–34.
18. Gustafson WH. Modules and matrices. *Linear Algebra and its Applications* November 1991; **157**:3–19, doi:10.1016/0024-3795(91)90100-B. URL `http://www.sciencedirect.com/science/article/B6V0R-45FKGCC-7B/2/40b933213ad066898d57a74cc5cf5284`.
19. Navasca C, Opperman M, Penderghest T, Tamon C. Tensors as module homomorphisms over group rings. *arXiv* 2010; **1005.1894**:1–11. URL `http://arxiv.org/abs/1005.1894`.
20. Kolda TG, Bader BW. Tensor decompositions and applications. *SIAM Review* August 2009; **51**(3):455–500, doi:10.1137/07070111X. URL `http://link.aip.org/link/?SIR/51/455/1`.
21. Qi L. Eigenvalues and invariants of tensors. *Journal of Mathematical Analysis and Applications* January 2007; **325**(2):1363 – 1377, doi:DOI:10.1016/j.jmaa.2006.02.071. URL `http://www.sciencedirect.com/science/article/B6WK2-4JK4PNS-5/2/4dcb0914917df6344bf71f709d1afe1f`.
22. Savas B, Eldén L. Krylov subspace methods for tensor computations. *Technical Report LiTH-MAT-R-2009-02-SE*, Linköpings University February 2009, doi:http://www.mai.liu.se/~laeld/tensorKrylov.pdf.
23. Savas B, Eldén L. Krylov-type methods for tensor computations I. *Linear Algebra and its Applications* 2011; **Available online**:In press, doi:10.1016/j.laa.2011.12.007.
24. Brazell M, Li N, Navasca C, Tamon C. Tensor and matrix inversions with applications. *arXiv* 2011; **math.NA**:1109.3830v1. URL `http://arxiv.org/abs/1109.3830`.
25. Tee GJ. Eigenvectors of block circulant and alternating circulant matrices. *Research Letters in the Information and Mathematical Sciences* 2005; **8**:123–142. URL `http://iims.massey.ac.nz/research/letters/volume8/tee/tee.pdf`.
26. Baker JR. Macrotasking the singluar value decomposition of block circulant matrices on the Cray-2. *Proceedings of the 1989 ACM/IEEE conference on Supercomputing*, Supercomputing '89, ACM: New York, NY, USA, 1989; 243–247, doi:10.1145/76263.76289.
27. Sinnott D, Harrington R. Analysis and design of circular antenna arrays by matrix methods. *Antennas and Propagation, IEEE Transactions on* Sep 1973; **21**(5):610–614.
28. De Mazancourt T, Gerlic D. The inverse of a block-circulant matrix. *Antennas and Propagation, IEEE Transactions on* September 1983; **31**(5):808–810. URL `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1143132&tag=1`.
29. Vescovo R. Inversion of block-circulant matrices and circular array approach. *Antennas and Propagation, IEEE Transactions on* Oct 1997; **45**(10):1565–1567, doi:10.1109/8.633869.
30. Rezghi M, Eldén L. Diagonalization of tensors with circulant structure. *Linear Algebra and its Applications* 2011; **435**(3):422 – 447, doi:10.1016/j.laa.2010.03.032.

31. Stewart GW. *Eigensystems*, *Matrix Algorithms*, vol. 2. SIAM: Philadelphia, 2001. URL `http://books.google.com/books?id=Y3k5TNTnEmEC`.
32. Davis PJ. *Circulant matrices*. Wiley, 1979.
33. Higham NJ. *Functions of Matrices: Theory and Computation*. SIAM, 2008.
34. Granata J, Conner M, Tolimieri R. The tensor product: a mathematical programming language for FFTs and other fast DSP operations. *IEEE Signal Processing Magazine* January 1992; **9**(1):40–48, doi: 10.1109/79.109206.
35. Chu MT, Watterson JL. On a multivariate eigenvalue problem, part I: algebraic theory and a power method. *SIAM J. Sci. Comput.* September 1993; **14**:1089–1106, doi:10.1137/0914066. URL `http://portal.acm.org/citation.cfm?id=159930.159936`.
36. Saad Y. *Iterative Methods for Sparse Linear Systems*. 2nd edn., Society for Industrial and Applied Mathematics: Philadelphia, 2003.

A. THE CIRCULANT SCALAR MAGNITUDE

This section describes another operation we extended to the circulant algebra. Eventually, we replaced it with our ordering (Definition 4), which is more powerful as we justify below. However, it plays a role in our MATLAB package, and thus we describe the rationale for our choice of magnitude function here.

For scalars in $\mathbb{R}$, the magnitude is often called the absolute value. Let $\alpha, \beta \in \mathbb{R}$. The absolute value has the the property $|\alpha\beta| = |\alpha|\,|\beta|$. We have already introduced an absolute value function, however. Here, we wish to define a notion of magnitude that produces a scalar in $\mathbb{R}$ to indicate the size of an element. Such a function will have norm-like flavor because it must represent the aggregate magnitude of k values with a single real-valued number. Thus, finding a function to satisfy $\left|\underline{\alpha} \circ \underline{\beta}\right| = \left|\underline{\alpha}\right| \left|\underline{\beta}\right|$ exactly is not possible. Instead, we seek a function $g : \mathbb{K}_k \mapsto \mathbb{R}$ such that

1. $g(\underline{\alpha}) = 0$ if and only if $\underline{\alpha} = 0$,
2. $g(\underline{\alpha} \circ \underline{\beta}) \leq g(\underline{\alpha})g(\underline{\beta})$,
3. $g(\underline{\alpha} + \underline{\beta}) \leq g(\underline{\alpha}) + g(\underline{\beta})$.

The following result shows that there is a large class of such magnitude functions.

Result 1
Any sub-multiplicative matrix norm $\|\boldsymbol{A}\|$ defines a magnitude function $g(\underline{\alpha}) = \|\mathtt{circ}(\underline{\alpha})\|$.

This result follows because the properties of the function g are identical to the requirements of a sub-multiplicative matrix norm applied to $\mathtt{circ}(\underline{\alpha})$. Any matrix norm induced by a vector norm is sub-multiplicative. In particular, the matrix 1, 2, and ∞ norms are all sub-multiplicative. Note that for circulant matrices both the matrix 1 and ∞ norms are equal to the 1-norm of any row or column, i.e., $\|\mathtt{vec}(\alpha)\|_1$ is a valid magnitude. Surprisingly, the 2-norm of the vector of parameters, that is $\|\mathtt{vec}(\underline{\alpha})\|_2$, is not. For a counterexample, let $\underline{\alpha} = \{1\ 2\}, \underline{\beta} = \{2\ 4\}$. Then $\underline{\alpha} \circ \underline{\beta} = \{8\ 10\}$ and $\left\|\mathtt{vec}(\underline{\alpha} \circ \underline{\beta})\right\|_2 = \sqrt{164} > \|\mathtt{vec}(\underline{\alpha})\|_2 \left\|\mathtt{vec}(\underline{\beta})\right\|_2 = \sqrt{100}$. For many practical computations, we use the matrix 2-norm of $\mathtt{circ}(\underline{\alpha})$ as the magnitude function. Thus,

$$|\underline{\alpha}| \equiv \|\mathtt{circ}(\underline{\alpha})\|_2 = \|\mathtt{cft}(\underline{\alpha})\|_1 .$$

This choice has the following relationship with our ordering:

$$\mathtt{abs}(\underline{\alpha}) \leq \mathtt{abs}(\underline{\beta}) \qquad \Rightarrow \qquad |\underline{\alpha}| \leq \left|\underline{\beta}\right| .$$

We implement this operation as the `mag` function in our MATLAB package.

DOI: 10.1002/nla