TAIWANESE JOURNAL OF MATHEMATICS
Vol. 15, No. 4, pp. 1497-1525, August 2011
This paper is available online at http://www.tjm.nsysu.edu.tw/

# A GLOBAL ARNOLDI METHOD FOR LARGE NON-HERMITIAN EIGENPROBLEMS WITH SPECIAL APPLICATIONS TO MULTIPLE EIGENPROBLEMS

Congying Duan and Zhongxiao Jia*

**Abstract.** Global projection methods have been used for solving numerous large matrix equations, but nothing has been known on if and how this method can be proposed for solving large eigenproblems. In this paper, a global Arnold method is proposed for large eigenproblems. It computes certain F-Ritz pairs that are used to approximate some eigenpairs. The global Arnoldi method inherits convergence properties of the standard Arnoldi method applied to a larger matrix whose distinct eigenvalues are the eigenvalues of the original given matrix. As an application, assuming that $A$ is diagonalizable, we show that the global Arnoldi method is able to solve multiple eigenvalue problems. To be practical, we develop an implicitly restarted global Arnoldi algorithm with certain F-shifts suggested. In particular, this algorithm can be adaptively used to solve multiple eigenvalue problems. Numerical experiments show that the algorithm is efficient for the eigenproblem and is reliable for quite ill-conditioned multiple eigenproblems.

## 1. Introduction

Jbilou, Messaoudi and Sadok [16] propose a global projection method for solving matrix equations. One of the main ingredients of the global methods is the use of the Frobenius scalar product. Simply speaking, here "global" describes the algorithms with the F-inner product to be defined by (1). They present a global Arnoldi process that generates an F-orthonormal basis of a matrix Krylov subspace, and based on it they derive the global FOM and the global GMRES methods. Several

Received November 16, 2009, accepted February 23, 2010.
Communicated by Wen-Wei Lin.
2000 *Mathematics Subject Classification*: 65F15, 15A18.
*Key words and phrases*: Global Arnoldi process, Global Arnoldi method, F-orthonormal, Eigenvalue problem, Multiple, F-Ritz value, F-Ritz vector, Convergence, Implicit restart.
Supported by the National Basic Research Program of China 2011CB302400 and the National Science Foundation of China (Nos. 10771116 and 11071140).
*Corresponding author.

authors [13, 14, 30] propose some other global methods, for instance, the global versions of CG, SCG, CR and CMRH. Over the past several years, numerous global methods have been widely used for solving linear systems with multiple right-hand sides and matrix equations, e.g., Sylvester equations and Riccati equations [4, 17, 18, 24, 31]. These methods fall into category of global projection methods onto certain matrix Krylov subspaces. A convergence analysis on the global GMRES method is made in [5]. These global Krylov subspace algorithms appear effective when applied for solving the problems mentioned above. Other applications of the global Krylov subspace methods are in model reduction, especially MIMO systems [7, 8, 9, 15]. However, no global projection method has been proposed for large matrix eigenproblems hitherto. Whether or not a global projection method can be proposed for the eigenproblems and, if yes, how to develop a practical algorithm have not been known.

For large non-Hermitian eigenproblems, a major class of methods are orthogonal projection methods, which include the famous Arnoldi method [1, 26, 29, 34]. The Arnoldi method uses the Arnoldi process to construct an orthonormal basis of the Krylov subspace generated by a single starting vector and compute the Ritz pairs to approximate some of the eigenpairs of a large $n \times n$ matrix $A$. Assuming that $A$ is diagonalizable, however, it is known that the Arnoldi method itself is unable to determine the multiplicities of the required eigenvalues and the associated eigenspaces [19, 20, 21]. In order to figure out these problems, block versions of the Arnoldi method are proposed [2, 20, 23] that first use the block Arnoldi process to construct an orthonormal basis of the block Krylov subspace generated by a set of vectors and then extract the Ritz pairs from the block Krylov subspaces to approximate the desired eigenpairs.

In this paper, based on the global Arnoldi process starting with an $n \times s$ initial matrix, we show how to derive a global Arnoldi method for large unsymmetric eigenproblems and set up a general framework of global projection methods for eigenproblems, also called the F-orthogonal projection methods. The method computes so-called F-Ritz pairs to approximate some of the eigenpairs of $A$. A fundamental difference with a usual projection method is that now there are $s$ F-Ritz vectors associated with each F-Ritz value, each of which can be used as an approximate eigenvector. So we can pick up any one of the F-Ritz vectors for our use for each F-Ritz value! We show that the F-Ritz values are equal to certain usual Ritz values of a larger matrix with each eigenvalue of $A$ as an $s$ multiple one over a closely related Krylov subspace and the F-Ritz pairs are at least as accurate as the usual Ritz pairs. So the global Arnoldi method inherits convergence properties of the standard Arnoldi method. We will pay special attention to the multiple eigenvalue problem. Under the assumption that $A$ is diagonalizable, we show that the global Arnoldi method can be applied to adaptively determine multiplicities of the desired

eigenvalues and the corresponding eigenspaces.

To be clearer and more illustrative, we state more on the global method. The global Arnoldi process constructs an F-orthonormal basis $V_1, V_2, \ldots, V_m$ of the matrix Krylov subspace $\mathcal{K}_m(A, V_1)$ generated by an $n \times s$ initial matrix $V_1$ with the Frobenius norm one, where $V_i = (v_{i1}, v_{i2}, \ldots, v_{is})$, $i = 1, 2, \ldots, m$ are $n \times s$ matrices. If we interpret the basis $(V_1, V_2, \ldots, V_m)$ as $ms$ linearly independent vectors, this matrix Krylov subspace can be regarded as a usual block Krylov subspace of dimension $ms$ starting with the initial block vector $V_1$, so that we can decompose it into the direct sum of the $s$ single vector Krylov subspaces $\mathcal{K}_m(A, v_{1j}),\ j = 1, 2, \ldots, s$ of dimension $m$. By the F-orthogonal projection principle, we can get $m$ approximate eigenvalues $\lambda_i^{(m)}$, $i = 1, 2, \ldots, m$, called the F-Ritz values with respect to the matrix Krylov subspace. For each F-Ritz value $\lambda_i^{(m)}$, we can obtain an approximate eigenvector $\varphi_{ij}^{(m)},\ j = 1, 2, \ldots, s$ from each single vector Krylov subspace. Assume that the F-Ritz value $\lambda_i^{(m)}$ and these $s$ corresponding F-Ritz vectors $\varphi_{i1}^{(m)}, \varphi_{i2}^{(m)}, \ldots, \varphi_{is}^{(m)}$ have converged. Then they are all good approximations to the eigenpairs of $A$. If a required eigenvalue $\lambda_i$ is simple, the $s$ F-Ritz vectors are linearly dependent numerically. So if the multiplicity of the desired eigenvalue is not concerned, we simply use any one of the $s$ F-Ritz vectors as an approximate eigenvector rather than compute all of them. If $\lambda_i$ is $d_i$ $(d_i < s)$ multiple, the $s$ F-Ritz vectors must be numerically linearly dependent, from which we can numerically determine the multiplicity $d_i$ reliably and efficiently. If $\lambda_i$ is $d_i$ $(d_i \geq s)$ multiple, these $s$ F-Ritz vectors are linearly independent. So $\lambda_i$ is at least $s$ multiple. We then run the global Arnoldi method starting with a new $V_1$ that are independent of the old $V_1$, and compute the new converged F-Ritz vectors. Adding them to the previous $\varphi_{ij}^{(m)},\ j = 1, 2, \ldots, s$, if they are linearly dependent numerically, then the rank is $d_i$; otherwise, continue. Proceed in such a way until $d_i$ is determined. We will quantitatively establish a solid foundation for the above claims. Both theory and numerical experiments illustrate that the procedure is reliable to determine eigenvalue multiplicities and eigenspaces when condition numbers of the desired eigenvectors are considerably smaller than reciprocals of the residual norms. This means that the procedure is effective for quite ill-conditioned multiple eigenproblems.

The global Arnoldi method becomes very expensive in storage and computational cost as $m$ increases. Therefore, restarting is necessary when the method does not deliver the approximate eigenpairs with prescribed accuracy for a maximum $m$ allowed. The need for restart was firstly recognized by Karush [22], and there are various restarting schemes developed by many researchers, e.g., Paige [25], Cullum and Donath [10], Golub and Underwood [12], Saad [27, 28] and Chatelin and Ho [6]. All of these schemes are explicit restarting. Over the years, the most popular restarting scheme is implicit restarting proposed by Sorensen [33] that

combines the implicitly shifted QR iterations with the Arnoldi process and leads to a truncated form of the implicitly shifted QR iteration. In this paper, we extend implicit restarting to the global Arnoldi process and develop an implicitly restarted global Arnoldi algorithm (IRGA) with those unwanted F-Ritz values as shifts, also called exact shifts as in [33].

The rest of the paper is organized as follows. In the next section, we briefly review the global Arnoldi process and the global FOM and GMRES algorithms. In Section 3, we propose a global Arnoldi method for large unsymmetric eigenproblems. In Section 4, we show how the global Arnoldi method is used to solve the multiple eigenproblem. In Section 5, we discuss how to implicitly restart the global Arnoldi method and develop the implicitly restarted global Arnoldi algorithm with the exact shifts suggested. Finally, we report numerical examples to illustrate efficiency and reliability of the algorithm in Section 6.

Some notations to be used are introduced. $A$ is an $n \times n$ large diagonalizable matrix throughout the paper, and $\lambda_i, \varphi_i$ are its eigenvalues and eigenvectors with $\lambda_i$ labeled in a desired order. Denote by $\|\cdot\|$ the spectral norm of a matrix and the vector 2-norm, by $\|\cdot\|_F$ the Frobenius norm of a matrix, by the superscript $H$ the conjugate transpose of a matrix or vector and by $I$ the identity matrix with the order clear from the context. The eigenvectors and their approximations are normalized to have unit length.

## 2. The Global Arnoldi Process and the Global FOM and GMRES

Let $\mathcal{M}_{n,s}$ denote the complex linear space of $n \times s$ rectangular matrices. For two matrices $X$ and $Y$ in $\mathcal{M}_{n,s}$, we define their F-inner product by

$$\langle X, Y\rangle_F = tr(X^H Y), \tag{1}$$

where $tr(Z)$ denotes the trace of the square matrix $Z$. Note that $\|\cdot\|_F = \langle\cdot\rangle_F^{1/2}$. We will use the notation $X \perp_F Y$ to denote $\langle X, Y\rangle_F = 0$, meaning that $X$ and $Y$ are F-orthogonal.

For a starting matrix $V \in \mathcal{M}_{n,s}$, the matrix Krylov subspace $\mathcal{K}_m(A, V)$ is defined by

$$\mathcal{K}_m(A, V) = span\{V, AV, \ldots, A^{m-1}V\}, \tag{2}$$

which is a subset of $\mathcal{M}_{n,s}$. $Z \in \mathcal{K}_m(A, V)$ means that there are scalars $\alpha_i$, $i = 1, 2, \ldots, m$ such that

$$Z = \sum_{i=0}^{m-1} \alpha_i A^i V. \tag{3}$$

Let $V = (v_1, v_2, \ldots, v_s)$ and define a linear operator $vec$: $\mathcal{M}_{n,s} \to \mathcal{C}^{ns}$ by

$$vec(V) = (v_1^H, v_2^H, \ldots, v_s^H)^H. \tag{4}$$

Then we have

$$\langle X, Y\rangle_F = \langle vec(X), vec(Y)\rangle, \tag{5}$$

where $\langle \cdot, \cdot \rangle$ denotes the usual $l_2$ inner product of the complex vector space $\mathcal{C}^{ns}$.

Denote by $A \otimes B$ the Kronecker product of the matrices $A$ and $B$. The following basic properties hold, e.g., [11]:

1. $(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$.
2. $(A \otimes B)^H = A^H \otimes B^H$.
3. If $A \in \mathcal{C}^{n\times n}$, $X \in \mathcal{C}^{n\times m}$, then $vec(AX) = (I_m \otimes A)vec(X)$.
4. Each eigenvalue $\lambda_i$, $i = 1, 2, \ldots, n$ of $A$ is an $s$ multiple eigenvalue of $I_s \otimes A$.

The following global Arnoldi process is based on the modified Gram-Schmidt process. It constructs an F-orthonormal basis $V_1, V_2, \ldots, V_m$ of the matrix Krylov subspace $\mathcal{K}_m(A, V)$, that is, $tr(V_i^H V_j) = \delta_{ij}$ for $i, j = 1, \ldots, m$, where $\delta_{ij}$ is the Kronecker delta.

**Algorithm 1.** The global Arnoldi process

1. $V_1 = V/\|V\|_F$
2. $for\ \ j = 1, 2, \ldots, m\ do$
   $W := AV_j$;
   $for\ \ i = 1, 2, \ldots, j\ do$
   $h_{i,j} = \langle W, V_i\rangle_F$;
   $W = W - h_{i,j}V_i$;
   $end$
   $h_{j+1,j} = \|W\|_F$;
   $V_{j+1} = W/h_{j+1,j}$.
   $end$

This process needs $ms$ matrix $A$ by vector products plus $nm^2s$ flops.

Define $\mathcal{V}_m = (V_1, V_2, \ldots, V_m)$ and $H_m$ and $\bar{H}_m$ to be the $m \times m$ and the $(m+1) \times m$ Hessenberg matrices whose nonzero entries $h_{i,j}$ are defined by Algorithm 1. Then

$$\bar{H}_m = \begin{pmatrix} H_m \\ h_{m+1,m}e_m^H \end{pmatrix}, \tag{6}$$

where $e_m = (0, \ldots, 0, 1)^H$ is the $m$-th canonical basis of $\mathcal{C}^m$. We have the following results [16, 24].

**Theorem 1.** *Let $\mathcal{V}_m$, $H_m$ and $\bar{H}_m$ be defined as above. Then $V_1, V_2, \ldots, V_m$ are an F-orthonormal basis of the matrix Krylov subspace $\mathcal{K}_m(A, V_1)$ and*

$$A\mathcal{V}_m = \mathcal{V}_m(H_m \otimes I_s) + h_{m+1,m}(0_{n\times s}, \ldots, 0_{n\times s}, V_{m+1}) \tag{7}$$

$$= \mathcal{V}_m(H_m \otimes I_s) + r_m(e_m^H \otimes I_s), \tag{8}$$

$$A\mathcal{V}_m = \mathcal{V}_{m+1}(\bar{H}_m \otimes I_s), \tag{9}$$

*where $0_{n\times s}$ denotes the $n \times s$ zero matrix, $r_m = h_{m+1,m}V_{m+1}$.*

Based on the global Arnoldi process, Jbilou *et al.* [16] propose a global FOM (GL-FOM) algorithm and a global GMRES (GL-GMRES) algorithm for solving linear systems with multiple right-hand sides and numerous matrix equations.

Taking the linear system with $s$ multiple right-hand sides as example, we briefly describe the GL-FOM and GL-GMRES algorithms. Let $X_0$ be an initial guess to the solution $X$ of $AX = B$ and $R_0 = B - AX_0$ its associated residual. At the $m$-th iterate of the Gl-FOM algorithm, the correction $Z_m = \mathcal{V}_m(y_m \otimes I_s)$ is extracted from the matrix Krylov subspace $\mathcal{K}_m(A, R_0)$ such that the residual $R_m = B - A(X_0 + Z_m) = R_0 - AZ_m$ satisfies

$$R_m \perp_F \mathcal{K}_m(A, R_0), \tag{10}$$

where $y_m$ is shown to be the solution of the following $m \times m$ linear system:

$$H_m y_m = \beta e_1, \quad \text{where } \beta = \|R_0\|_F. \tag{11}$$

In the GL-GMRES algorithm, the correction $Z_m = \mathcal{V}_m(y_m \otimes I_s)$ is determined by imposing the residual minimization condition

$$\|R_m\|_F = \min_{y\in\mathcal{C}^m} \|R_0 - A\mathcal{V}_m(y \otimes I_s)\|_F, \tag{12}$$

where $y_m$ is shown to be the solution of the $(m+1) \times m$ least squares problem

$$y_m = arg \min_{y\in\mathcal{C}^m} \|\beta e_1 - \bar{H}_m y\|, \quad \text{where } \beta = \|R_0\|_F. \tag{13}$$

For details on these two global methods, we refer to [16].

We remark that if $s = 1$ then the global Arnoldi process reduces to the standard Arnoldi process and the GL-FOM and GL-GMRES methods become the standard FOM (Arnoldi) and GMRES methods.

## 3. A Global Arnoldi Method for Eigenproblems

As a first step towards deriving a global Arnoldi method for eigenproblems, a simple but fundamental key is that we can interpret the matrix Krylov subspace $\mathcal{K}_m(A, V_1)$ of $\mathcal{M}_{n,s}$ as a standard block Krylov subspace of $\mathcal{C}^n$ starting with the

initial block vector $V_1$. Correspondingly, we can interpret each basis element $V_i, i = 1, 2, \ldots, m$ of the matrix Krylov subspace $\mathcal{K}_m(A, V_1)$ as usual $s$ vectors and each $n \times s$ element of it as $s$ vectors instead of an $n \times s$ matrix. We will switch $\mathcal{K}_m(A, V_1)$ between the matrix Krylov subspace and the standard Krylov subspace, following our need. In the sequel, when speaking of the block Krylov subspace, suppose that $\mathcal{V}_m = (V_1, V_2, \ldots, V_m)$ is of full column rank. Then Algorithm 1 also generates a basis of the block Krylov subspace $\mathcal{K}_m(A, V_1)$ in $\mathcal{C}^n$. However, we should keep in mind that this basis is generally not orthogonal.

Mathematically, when $\mathcal{K}_m(A, V_1)$ is regarded as the block Krylov subspace, we might use the standard orthogonal projection principle to solve eigenproblems. In this case, premultiplying (7) by $(\mathcal{V}_m^H \mathcal{V}_m)^{-1} \mathcal{V}_m^H$ gives

$$\begin{aligned} &(\mathcal{V}_m^H \mathcal{V}_m)^{-1} \mathcal{V}_m^H A \mathcal{V}_m \\ &= H_m \otimes I_s + (\mathcal{V}_m^H \mathcal{V}_m)^{-1} \mathcal{V}_m^H h_{m+1,m}(0_{n\times s}, \ldots, 0_{n\times s}, V_{m+1}), \end{aligned} \tag{14}$$

which is just the projection matrix of $A$ onto the subspace spanned by the columns of $\mathcal{V}_m$, whose $ms$ eigenvalues $\hat{\lambda}_i^{(m)}$ are just the Ritz values of $A$ with respect to this subspace and the (unnormalizing) Ritz vectors are $\hat{\varphi}_i^{(m)} = \mathcal{V}_m \hat{y_i}^{(m)}$, where the $\hat{y}_i^{(m)}$ are the eigenvectors of the projection matrix associated with the eigenvalues $\hat{\lambda}_i^{(m)}$. Mathematically, they are the same as the Ritz values and Ritz vectors when the standard block Arnoldi process is exploited to generate an orthonormal basis of $\mathcal{K}_m(A, V_1)$. However, computationally, we do not recommend this approach. Firstly, it may be unstable as the columns of $\mathcal{V}_m$ can be nearly linearly dependent so that $\mathcal{V}_m^H \mathcal{V}_m$ is nearly singular; secondly, it is more costly than the block Arnoldi method for the same $m$. Actually, apart from the cost of $m$-step global Arnoldi process, we need $n(ms)^2 + 2(ms)^3$ flops for forming $\mathcal{V}_m^H \mathcal{V}_m$ and inverting it, while the block Arnoldi process costs $ms$ matrix by vector products plus $n(ms)^2$ flops, assuming that no reorthogonalization is used. So such a procedure costs more than the block Arnoldi process for the same $m$. Therefore, we have to abandon it and seek other viable procedures.

Define $\mathcal{A} = I_s \otimes A$. Then it has each eigenvalue $\lambda_i$ of $A$ as an $s$ multiple eigenvalue. Assume that $A$ is diagonalizable and define $D = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$ and the eigenvector matrix

$$\Phi = (\varphi_1, \varphi_2, \ldots, \varphi_n).$$

Then

$$\begin{pmatrix} \Phi & & & \\ & \Phi & & \\ & & \ddots & \\ & & & \Phi \end{pmatrix}^{-1} \mathcal{A} \begin{pmatrix} \Phi & & & \\ & \Phi & & \\ & & \ddots & \\ & & & \Phi \end{pmatrix} = \begin{pmatrix} D & & & \\ & D & & \\ & & \ddots & \\ & & & D \end{pmatrix}.$$

From this eigen-decomposition, we get the $s$ corresponding eigenvectors of $\mathcal{A}$ associated with $\lambda_i$ that have the form $w_i = (0, \ldots, \varphi_i^H, \ldots, 0)^H$, whose possible nonzero entries are in positions $n(i-1)+1$ to $ni$, $i = 1, 2, \ldots, s$. Any linear combination of them is still an eigenvector of $\mathcal{A}$ associated with $\lambda_i$ and may have no special structure.

The global Arnoldi process on $A$ starting with the matrix $V_1$ is closely related to the standard Arnoldi process on $\mathcal{A}$ starting with the initial vector $vec(V_1)$. The following results can be easily justified [24], and they are the very first step to propose and understand a global Arnoldi method for eigenproblems.

**Theorem 2.** *Let $H_m$ and $\bar{H}_m$ be defined as above. Then $vec(V_1), vec(V_2), \ldots, vec(V_m)$ form an orthonormal basis of the usual Krylov subspace $\mathcal{K}_m\big(\mathcal{A}, vec(V_1)\big)$ generated by $\mathcal{A}$ and the starting vector $vec(V_1)$. Define the matrix*

$$\mathbf{V}_m = \big(vec(V_1), vec(V_2), \ldots, vec(V_m)\big).$$

*Then the following standard Arnoldi process holds:*

$$\mathcal{A}\mathbf{V}_m = \mathbf{V}_m H_m + h_{m+1,m} vec(V_{m+1}) e_m^H, \tag{15}$$

$$= \mathbf{V}_m \bar{H}_m. \tag{16}$$

Since $vec(V_1), vec(V_2), \ldots, vec(V_m)$ are orthonormal, Theorem 2 shows that $H_m$ is the orthogonal projection matrix of $\mathcal{A}$ onto the subspace $\mathcal{K}_m\big(\mathcal{A}, vec(V_1)\big)$. Let $\lambda_i^{(m)}, y_i^{(m)}$, $i = 1, 2, \ldots, m$ be the eigenpairs of $H_m$ with $\|y_i^{(m)}\| = 1$. The eigenvalues $\lambda_i^{(m)}$ are the Ritz values of $\mathcal{A}$ with respect to the subspace $\mathcal{K}_m(\mathcal{A}, vec(V_1))$ and the corresponding Ritz vectors are

$$w_i^{(m)} = (vec(V_1), vec(V_2), \ldots, vec(V_m)) y_i^{(m)},\ i = 1, 2, \ldots, s.$$

So we can simply use the Ritz pairs to approximate some eigenpairs of $\mathcal{A}$. The residual norms are computed cheaply by

$$\|\mathcal{A} w_i^{(m)} - \lambda_i^{(m)} w_i^{(m)}\| = h_{m+1m} \mid e_m^H y_i^{(m)} \mid \tag{17}$$

without explicitly forming $w_i^{(m)}$ until the convergence occurs.

However, the situation is subtle and by no means so simple as $\mathcal{A}$ is much larger than $A$ in size and all the eigenvalues are at least $s$ multiple. We note that on one hand each eigenvalue of $A$ is an $s$ multiple one of $\mathcal{A}$ and on the other hand the eigenvalues of $H_m$ are always simple if it is diagonalizable. In fact, assuming that $A$ is diagonalizable, the standard Arnoldi method works as if $\mathcal{A}$ had only simple eigenvalues [19, 20, 21, 26, 27]. Therefore, when a Ritz pair defined above

converges, we can get only one simple approximation to the $s$ multiple eigenpairs of $\mathcal{A}$. So we have to do more.

We now propose a better and practical global Arnoldi method that works on the original $A$ directly rather than on the much larger $ns \times ns$ matrix $\mathcal{A}$. Recall that the global Arnoldi process constructs an F-orthonormal basis $V_1, V_2, \ldots, V_m$ of the matrix Krylov subspace $\mathcal{K}_m(A, V_1)$. When speaking of a block Krylov subspace $\mathcal{K}_m(A, V_1)$, we can simply decompose it into the direct sum of the $s$ single vector Krylov subspaces generated by the starting vectors $v_{11}, v_{12}, \ldots, v_{1s}$, respectively:

$$\mathcal{K}_m(A, V_1) = \mathcal{K}_m(A, v_{11}) \oplus \mathcal{K}_m(A, v_{12}) \oplus \cdots \oplus \mathcal{K}_m(A, v_{1s}). \tag{18}$$

Keeping in mind the assumption that the columns of $V_1, V_2, \ldots, V_m$ are linearly independent. Then in the MATLAB language the columns of $\mathcal{V}_m^j = \mathcal{V}_m(:, j : s : ms)$ form a basis of $\mathcal{K}_m(A, v_{1j})$, $j = 1, 2, \ldots, s$. Since $v_{1j}$, $j = 1, 2, \ldots, s$ are supposed to be linearly independent, we get the $s$ distinct $m$-dimensional Krylov subspaces $\mathcal{K}_m(A, v_{1j})$. Now we still use $\lambda_i^{(m)}$, $i = 1, 2, \ldots, m$ as approximations to some of the eigenvalues of $A$, called the F-Ritz values of $A$ with respect to the matrix Krylov subspace $\mathcal{K}_m(A, V_1)$, but we compute the $s$ new vectors

$$\varphi_{ij}^{(m)} = \mathcal{V}_m^j y_i^{(m)}, \quad j = 1, 2, \ldots, s, \tag{19}$$

called the F-Ritz vectors of $A$ with respect to $\mathcal{K}_m(A, V_1)$. For each $\lambda_i^{(m)}$ we now have the $s$ corresponding F-Ritz vectors $\varphi_{ij}^{(m)} \in \mathcal{K}_m(A, v_{1j})$.

We shed more lights on the F-Ritz pairs. Suppose that they already converge. Then if $\lambda_i$ is simple, these $s$ F-Ritz vectors must be almost parallel as they approximate the same $\varphi_i$. If $\lambda_i$ is multiple, each of these $s$ F-Ritz vectors is a good approximate eigenvector of $A$. If we do not care the multiplicity of $\lambda_i$ and do not determine the whole eigenspace of $A$ associated with $\lambda_i$, then we can simply use any of the F-Ritz vectors as an approximate eigenvector of $A$ instead of computing all of them.

Let $U_i^{(m)} = \mathcal{V}_m(y_i^{(m)} \otimes I_s) = (\mathcal{V}_m^1 y_i^{(m)}, \ldots, \mathcal{V}_m^s y_i^{(m)}) = (\varphi_{i1}^{(m)}, \ldots, \varphi_{is}^{(m)})$. Then $\|U_i^{(m)}\|_F = 1$ and it is easily verified that $(\lambda_i^{(m)}, U_i^{(m)})$, $i = 1, 2, \ldots, m$ are the solutions of

$$\begin{cases} U_i^{(m)} \in \mathcal{V}_m \\ (A - \lambda_i^{(m)} I) U_i^{(m)} \perp_F \mathcal{V}_m. \end{cases} \tag{20}$$

We call the $(\lambda_i^{(m)}, U_i^{(m)})$ the F-Ritz pairs of $A$ with respect to the matrix Krylov subspace $\mathcal{K}_m(A, V_1)$. Therefore, the global Arnoldi method falls into a general framework, and we call it the F-orthogonal projection.

We investigate how a F-Ritz value and a F-Ritz vector approximates an eigenvalue and its associated eigenvector of $A$ by relating it to the standard Ritz pair $(\lambda_i^{(m)}, w_i^{(m)})$ of $\mathcal{A}$ with respect to $\mathcal{K}_m\big(\mathcal{A}, vec(V_1)\big)$.

**Theorem 3.** *We have*

$$\|(A-\lambda_i^{(m)}I)U_i^{(m)}\|_F = h_{m+1,m} \mid e_m^H y_i^{(m)} \mid, \tag{21}$$

$$\|(A-\lambda_i^{(m)}I)\varphi_{ij}^{(m)}\| \leq h_{m+1,m} \mid e_m^H y_i^{(m)} \mid,\ j=1,2,\ldots,s. \tag{22}$$

*Proof.* From (7) we have

$$\begin{aligned}
&\|(A-\lambda_i^{(m)}I)U_i^{(m)}\|_F \\
&= A\mathcal{V}_m(y_i^{(m)}\otimes I_s) - \lambda_i^{(m)}\mathcal{V}_m(y_i^{(m)}\otimes I_s) \\
&= \mathcal{V}_m(H_m y_i^{(m)}\otimes I_s) + h_{m+1,m}(0_{n\times s},\ldots,0_{n\times s},V_{m+1})(y_i^{(m)}\otimes I_s) \\
&\quad -\lambda_i^{(m)}\mathcal{V}_m(y_i^{(m)}\otimes I_s) \\
&= h_{m+1,m}(0_{n\times s},\ldots,0_{n\times s},V_{m+1})(y_i^{(m)}\otimes I_s) \\
&= h_{m+1,m}V_{m+1}(e_m^H\otimes I_s)(y_i^{(m)}\otimes I_s) \\
&= h_{m+1,m}e_m^H y_i^{(m)}V_{m+1}.
\end{aligned}$$

Therefore, we get

$$\|(A-\lambda_i^{(m)}I)U_i^{(m)}\|_F = h_{m+1,m} \mid e_m^H y_i^{(m)} \mid,$$

which is just (21).

Noting that

$$\|A\mathcal{V}_m^j y_i^{(m)} - \lambda_i^{(m)}\mathcal{V}_m^j y_i^{(m)}\| \leq \|A\mathcal{V}_m(y_i^{(m)}\otimes I_s) - \lambda_i^{(m)}\mathcal{V}_m(y_i^{(m)}\otimes I_s)\|_F,$$

we have shown (22). ∎

Recall (17). (21) indicates that the residual Frobenius norm of the F-Ritz pair $(\lambda_i^{(m)}, U_i^{(m)})$ is just equal to the residual 2-norm of the Ritz pair $(\lambda_i^{(m)}, w_i^{(m)})$ obtained by the standard Arnoldi method applied to $\mathcal{A}$ over $\mathcal{K}_m(\mathcal{A}, vec(V_1))$.

(22) can be used as a stopping criterion which cheaply checks if the global Arnoldi method converges without computing $\varphi_{ij}^{(m)}$ by (19) until convergence occurs.

Recall that for the standard Arnoldi method on $\mathcal{A}$ over $\mathcal{K}_m(\mathcal{A}, vec(V_1))$ the Ritz vectors

$$w_i^{(m)} = \mathbf{V}_m y_i^{(m)} = [(\mathcal{V}_m^1)^H,\ldots,(\mathcal{V}_m^s)^H]^H y_i^{(m)} = [(\varphi_{i1}^{(m)})^H,\ldots,(\varphi_{is}^{(m)})^H]^H$$

with $\|w_i^{(m)}\| = 1$ as it is assumed that $\|y_i^{(m)}\| = 1$. So the $\varphi_{ij}^{(m)}$ are unnormalized and their norms are always smaller than one. Since no one of the non-orthonormal bases $\mathcal{V}_m^j$ is special, the $\|\varphi_{ij}^{(m)}\|$ should generally be comparable in

size, i.e., $\|\varphi_{ij}^{(m)}\| \approx \frac{1}{\sqrt{s}}$. Therefore, we get

$$\begin{aligned}
&\|A\mathcal{V}_m^j y_i^{(m)} - \lambda_i^{(m)}\mathcal{V}_m^j y_i^{(m)}\| \\
\approx\ & \frac{1}{\sqrt{s}}\|\left[(A\mathcal{V}_m^1 y_i^{(m)} - \lambda_i^{(m)}\mathcal{V}_m^1 y_i^{(m)}), \ldots, (A\mathcal{V}_m^s y_i^{(m)} - \lambda_i^{(m)}\mathcal{V}_m^s y_i^{(m)})\right]\|_F \\
=\ & \frac{1}{\sqrt{s}}\|A\mathcal{V}_m(y_i^{(m)} \otimes I_s) - \lambda_i^{(m)}\mathcal{V}_m(y_i^{(m)} \otimes I_s)\|_F \\
=\ & \frac{1}{\sqrt{s}} h_{m+1,m} \mid e_m^H y_i^{(m)} \mid,
\end{aligned}$$

So combining the above and the proof of Theorem 3, we have

$$\|r_{ij}^{(m)}\| := \frac{\|(A - \lambda_i^{(m)} I)\varphi_{ij}^{(m)}\|}{\|\varphi_{ij}^{(m)}\|} \approx h_{m+1,m} \mid e_m^H y_i^{(m)} \mid . \tag{23}$$

The left-hand side of the above relation is the residual norm of the normalized F-Ritz pair, while from (17) the right-hand side is just the residual norm of $(\lambda_i^{(m)}, w_i^{(m)})$ as an approximate eigenpair of $\mathcal{A}$. So, (22) and (23) demonstrates that all the $s$ F-Ritz vectors $\varphi_{ij}^{(m)}$ are good approximate eigenvectors of $A$ associated with the eigenvalue $\lambda_i$ if the standard Arnoldi method applied to $\mathcal{A}$ converges. So the global Arnoldi method inherits convergence properties of the standard Arnoldi method and achieve comparable residuals for the same $m$. For a convergence analysis of the standard Arnoldi method, we refer to [19, 20, 21, 26, 27].

We now present a basic global Arnoldi algorithm for eigenproblems.

**Algorithm 2.** *A basic global Arnoldi algorithm*

1. Let $(\lambda_i, \varphi_i)$, $i = 1, 2, \ldots, k$ be $k$ desired eigenpairs of $A$ and $tol$ a user prescribed tolerance. Choose an $n \times s$ matrix $V$ and take $V_1 = V/\|V\|_F$ as the starting matrix.
2. For $m = k+1, k+2, \ldots$ until convergence
   (a) Construct the F-orthonormal basis $V_1, V_2, \ldots, V_m$ by Algorithm 1.
   (b) Compute the $m$ eigenpairs $(\lambda_i^{(m)},\ y_i^{(m)})$, $i = 1, 2, \ldots, m$ of the resulting Hessenberg matrix $H_m$ and select $k$ F-Ritz values, say, $\lambda_i^{(m)}$, $i = 1, 2, \ldots, k$ as approximations to the $k$ desired $\lambda_i$, $i = 1, 2, \ldots, k$.
   (c) Form $\varphi_i^{(m)} = \varphi_{i1}^{(m)} = \mathcal{V}_m^1 y_i$, $i = 1, 2, \ldots, k$.
   (d) Test convergence of the $k$ approximate eigenpairs $(\lambda_i^{(m)}, \varphi_i^{(m)})$.
   (e) If they all drop below tol, then go to Step 3.

We should mention that if $s = 1$ then the global Arnoldi method is just the standard basic Arnoldi method.

## 4. Multiple Eigenproblems

As we have seen previously, under the assumption that $A$ is diagonalizable, if $H_m$ is diagonalizable, F-Ritz values are always simple even if $A$ has multiple eigenvalues. So the global Arnoldi method works as if $A$ has only simple eigenvalues. As a result, when a desired $\lambda_i$ is multiple, the method itself cannot determine the multiplicity of $\lambda_i$ and compute the eigenspace associated with it.

Exploiting the theoretical analysis of [21], we consider how the global Arnoldi method can be successfully used to solve multiple eigenvalue problems. Before discussions, we need some notations. We always assume that $A$ is an $n \times n$ diagonalizable matrix and has $M$ distinct eigenvalues $\lambda_i$, where the multiplicities of $\lambda_i$ are $d_i$, $i = 1, 2, \ldots, M$. Let $\mathcal{P}_i$ be the $d_i$-dimensional eigenspace associated with $\lambda_i$ and the columns of $\Phi_{id_i} = (\varphi_{i1}, \varphi_{i2}, \ldots, \varphi_{id_i})$ form a basis of $\mathcal{P}_i$, where $\|\varphi_{ij}\| = 1, j = 1, 2, \ldots, d_i$.

Write the $n \times s$ starting matrix $V_1$ as $V_1 = (v_{11}, v_{12}, \ldots, v_{1s})$. Then given $\Phi_{id_i}$, each $v_{1j}, 1 \le j \le s$, can be uniquely expanded as

$$\begin{aligned} &v_{1j} = b_{j1}\varphi_{i1} + b_{j2}\varphi_{i2} + \cdots + b_{jd_i}\varphi_{id_i} + u_{ij}, \\ &u_{ij} \in \mathcal{P}_1 \oplus \cdots \oplus \mathcal{P}_{i-1} \oplus \mathcal{P}_{i+1} \oplus \cdots \oplus \mathcal{P}_M. \end{aligned} \tag{24}$$

Define

$$B_s = \begin{pmatrix} b_{11} & b_{12} & \ldots & b_{1d_i} \\ b_{21} & b_{22} & \ldots & b_{2d_i} \\ \vdots & \vdots & \ldots & \vdots \\ b_{s1} & b_{s2} & \ldots & b_{sd_i} \end{pmatrix}. \tag{25}$$

Obviously, $B_s$ is row rank deficient when $s > d_i$. Assume that the matrix $B_s$ is of row full rank for $s \le d_i$. We rewrite the above $v_{1j}$ as

$$v_{1j} = \beta_j \tilde{\varphi}_{ij} + u_{ij},\ u_{ij} \in \mathcal{P}_1 \oplus \cdots \oplus \mathcal{P}_{i-1} \oplus \mathcal{P}_{i+1} \oplus \cdots \oplus \mathcal{P}_M, \tag{26}$$

where $\tilde{\varphi}_{ij}$, $j = 1, 2, \ldots, s$ are also unit length eigenvectors associated with $\lambda_i$. Under the assumption on $B_s$, just as $\{\varphi_{ij}\}_{j=1}^{d_i}$, $\{\tilde{\varphi}_{ij}\}_{j=1}^{d_i}$ is also a basis of $\mathcal{P}_i$, and for $s > d_i$, $\tilde{\varphi}_{ij}$, $j = d_i+1, \ldots, s$ must be linearly dependent to $\tilde{\varphi}_{ij}$, $j = 1, 2, \ldots, d_i$ and belong to the span of $\{\tilde{\varphi}_{ij}\}_{j=1}^{d_i}$. In other words, define $\tilde{\Phi}_{is} = (\tilde{\varphi}_{i1}, \tilde{\varphi}_{i2}, \ldots, \tilde{\varphi}_{is})$ and $\tilde{\Phi}_{is}^{(m)} = (\tilde{\varphi}_{i1}^{(m)}, \tilde{\varphi}_{i2}^{(m)}, \ldots, \tilde{\varphi}_{is}^{(m)})$. Then $\tilde{\Phi}_{is}$ is of full column rank for $s \le d_i$, while it must be column rank deficient and the smallest singular value of it is zero for $s > d_i$. From now on we omit the tilde in the Greek letters without ambiguity; furthermore, we assume to have normalized $\varphi_{ij}^{(m)}$ in the discussions below. We also assume that the columns of $\Phi_{is}$ are strongly linear independent for $s \le d_i$, that is, the smallest singular value of $\Phi_{ij}$ is not small and moderate for $s \le d_i$. From

the viewpoint of the probability theory, this assumption is holds for a basis of $\mathcal{P}_i$ generated randomly.

The following theorem from [21] is the theoretical background for determining $d_i$ numerically by the global Arnoldi method.

**Theorem 4.** *Let $P_1, \ldots, P_M$ be the spectral projectors associated with $\lambda_1, \ldots, \lambda_M$, and define the matrix*

$$X_{ij} = \left( \frac{P_1 \varphi_{ij}^{(m)}}{\|P_1 \varphi_{ij}^{(m)}\|}, \ldots, \frac{P_{i-1} \varphi_{ij}^{(m)}}{\|P_{i-1} \varphi_{ij}^{(m)}\|}, \frac{P_{i+1} \varphi_{ij}^{(m)}}{\|P_{i+1} \varphi_{ij}^{(m)}\|}, \ldots, \frac{P_M \varphi_{ij}^{(m)}}{\|P_M \varphi_{ij}^{(m)}\|} \right)$$

*and $g_i = \min_{k \neq i} |\lambda_i^{(m)} - \lambda_k|$. Then*

$$C_{ij} = \frac{\kappa(X_{ij}) \|I - P_i\|}{g_i} \leq \frac{\kappa(X_{ij})(1 + \|P_i\|)}{g_i}, \tag{27}$$

$$\sin \angle(\varphi_{ij}, \varphi_{ij}^{(m)}) \leq C_{ij} \|r_{ij}^{(m)}\|, \tag{28}$$

*where $\kappa(X_{ij})$ is the spectral condition number of $X_{ij}$ and $r_{ij}^{(m)}$ is the residual defined by (23). Let $\sigma_{\min}(\Phi_{is}^{(m)})$ and $\sigma_{\min}(\Phi_{is})$ be the smallest singular values of the matrices $\Phi_{is}^{(m)}$ and $\Phi_{is}$, respectively. Then*

$$\sigma_{\min}(\Phi_{is}^{(m)}) \leq \sigma_{\min}(\Phi_{is}) + \sqrt{s} \max_{1 \leq j \leq s} \|\varphi_{ij} - \varphi_{ij}^{(m)}\|. \tag{29}$$

*In particular, if $s > d_i$, then*

$$\sigma_{\min}(\Phi_{is}^{(m)}) \leq \sqrt{s} \max_{1 \leq j \leq s} \|\varphi_{ij} - \varphi_{ij}^{(m)}\| \tag{30}$$

$$\approx \sqrt{s} \cdot C_{ij} \max_{1 \leq j \leq s} \|r_{ij}^{(m)}\| \quad \textit{for small} \ \ \|r_{ij}^{(m)}\|. \tag{31}$$

The relation (28) estimates the accuracy of $\varphi_{ij}^{(m)}$ in terms of the residual norm $\|r_{ij}^{(m)}\|$, where $C_{ij}$ acts as a condition number and measures the conditioning of $\varphi_{ij}$. The bigger it is, the worse conditioned $\varphi_{ij}$ is. If one of $\kappa(X_{ij})$ and $\|P_i\|$ is big or the separation $g_i$ of the approximate $\lambda_i^{(m)}$ and the other exact eigenvalues $\lambda_j$ is very small, $\varphi_{ij}$ is ill conditioned. We can see from (31) that if $C_{ij}$ is comparable to or bigger than $\frac{1}{\max_{1 \leq j \leq s} \|r_{ij}^{(m)}\|}$ then $\sigma_{\min}(\Phi_{is}^{(m)})$ may not be small.

Based on this theorem, we can decide if $\Phi_{is}^{(m)},\ i = 1, 2, \ldots, k$, are approximately column rank deficient in the sense of (31) and thus detect the multiplicities $d_i$ and get an approximate basis of $\mathcal{P}_i$. It tells us that $\Phi_{is}^{(m)},\ i = 1, 2, \ldots, k$ are approximately

column rank deficient for $d_i < s$ and have full column rank for $d_i \geq s$. So $\sigma_{\min}(\Phi_{is}^{(m)}),\ i = 1, 2, \ldots, k$ are not small for $d_i \geq s$. We decide $d_i$ in such a way: Assume that $\|r_{ij}^{(m)}\| < tol$. Then if $s$ is the smallest integer such that

$$\sigma_{\min}(\Phi_{is}^{(m)}) \leq \sqrt{s} \cdot C_{ij} \max_{1 \leq j \leq s} \|r_{ij}^{(m)}\| \tag{32}$$

hold with a constant $C_{ij}$ significantly smaller than $\max\limits_{1 \leq j \leq s} \|r_{ij}^{(m)}\|$, $\lambda_i$ is $(s-1)$ multiple and $d_i = s - 1$. In practice, $C_{ij}$ is a-priori unknown. We take $C_{ij}$ to be considerably less than $\frac{1}{tol}$, say $\frac{10^{-3}}{tol}$ or smaller, which means that $\varphi_{ij}, j = 1, 2, \ldots, s$ can be quite ill conditioned, such as $C_{ij} = 10^3$ or $10^5$ if $\max\limits_{1 \leq j \leq s} \|r_{ij}^{(m)}\| \leq 10^{-6}$ or $10^{-8}$. So the procedure may fail to determine $d_i$ if $C_{ij}$ is comparable to or bigger than $\frac{1}{tol}$ but it is definitely reliable. Later numerical experiments will indeed illustrate that (31) is conservative and our procedure can determine the multiplicities of eigenvalues for quite ill-conditioned eigenproblems, e.g., $C_{ij} \geq 4.5 \times 10^4$.

In practice, $s$ is given. A random $V_1$ will make $B_s$ defined by (25) satisfy the assumption on the rank of it. With $V_1$, if we compute an $s$ numerically multiple eigenvalue, then this eigenvalue is at least $s$ multiple, so the determination of $d_i$ is not actually resolved when $s \leq d_i$. The following approach taken from [20, 21] can figure out this problem elegantly.

Firstly, choose a random starting matrix $V_1^{(1)}$ with $s_1$ columns. If we have found an $s_1$ multiple eigenvalue, then we apply Algorithm 2 with a new starting initial $V_1^{(2)}$ with $s_2$ columns, which is chosen randomly. Now we can compute an $s_2 \leq s_1$ multiple eigenvalue, which is numerically equal to the one computed with $s_1, V_1^{(1)}$. We then determine the rank of the matrix $(\Phi_{is_1}^{(m)}, \Phi_{is_2}^{(m)})$ consisting of these $s_1 + s_2$ converged F-Ritz vectors with the numerically multiple converged F-Ritz values. Note here that when the eigenproblem of $A$ is not too ill conditioned, if some singular values of this matrix are of the same order as the maximum of residual norms of these $s_1 + s_2$ converged interior eigenpairs, then we consider them to be zero numerically. If the numerical rank of the matrix is less than $s_1 + s_2$, then the multiplicity of this eigenvalue is just the rank of such a matrix. Otherwise, we repeat Algorithm 2 with $s_3 \leq s_1$, $V_1^{(3)}$ and so on until the numerical rank of the matrix consisting of these $s_1 + s_2 + \cdots + s_q$ converged F-Ritz vectors starting with $s_1, V_1^{(1)}, s_2, V_1^{(2)}, \ldots, s_q, V_1^{(q)}$ respectively, is smaller than $s_1 + s_2 + \cdots + s_q$. Then the multiplicity $d_i$ of the eigenvalue has been determined and equals the numerical rank of this matrix.

In summary, we can present a global Arnoldi algorithm for multiple eigenproblems.

**Algorithm 3.** A global Arnoldi algorithm for multiple eigenproblems

1. Define the set $\mathcal{S}=\{1,2,\ldots,k\}$, $\Psi=\emptyset$, $q=1$ and prescribe a tolerance tol.
2. Choose a starting $n \times s_q$ matrix $V_1$ with $\|V_1\|_F = 1$.
3. For $m = k+1, k+2, \ldots$ until convergence
   (a) Construct the F-orthonormal basis $V_1, V_2, \ldots, V_m$ by Algorithm 1.
   (b) Compute the $m$ eigenpairs $(\lambda_i^{(m)}, y_i^{(m)})$, $i = 1, 2, \ldots, m$ of the resulting Hessenberg matrix $H_m$ and use $\lambda_i^{(m)}, i = 1, 2, \ldots, k$ to approximate the desired $\lambda_i$, $i = 1, 2, \ldots, k$.
   (c) Test convergence of the approximate eigenpairs $(\lambda_i^{(m)}, \varphi_{ij}^{(m)})$, $i = 1, 2, \ldots, k$; $j = 1, 2, \ldots, s_q$.
   (d) If they all drop below tol, then go to Step 4.
4. For all $i \in \mathcal{S}$, set $\Phi_{is}^{(m)} = (\Psi, \varphi_{i1}^{(m)}, \varphi_{i2}^{(m)}, \ldots, \varphi_{is_q}^{(m)})$ and $s$ the number of its columns.
   (a) Compute the numerical rank $r_s$ of $\Phi_{is}^{(m)}$ for all $i \in \mathcal{S}$;
   (b) If $r_s < s$, set $d_i = r_s$ and remove $i$ from $\mathcal{S}$;
   (c) Otherwise, let $\Psi = \Phi_{is}^{(m)}, q = q+1$ and go to Step 2.

## 5. An Implicitly Restarted Global Arnoldi Algorithm

The basic global Arnoldi algorithm becomes very expensive and impractical due to excess storage and high computational cost as $m$ increases. So $m$ must be limited not to be big. To make the method practical, restarting is necessary. The implicit restarting technique due to Sorensen [33] is a very successful and popular one. We show how to extend it to the global Arnoldi process and develop an implicitly restarted global Arnoldi algorithm (IRGA).

Let $k$ be a fixed specified integer, usually the number of the desired eigenpairs of $A$ and the steps $m = k + p$. Consider the $k+p$ step global Arnoldi process

$$A\mathcal{V}_{k+p} = \mathcal{V}_{k+p}(H_{k+p} \otimes I_s) + r_{k+p}(e_{k+p}^H \otimes I_s) \tag{33}$$

$$= (\mathcal{V}_{k+p}, V_{k+p+1})\left(\begin{pmatrix} H_{k+p} \\ \beta_{k+p} e_{k+p}^H \end{pmatrix} \otimes I_s\right). \tag{34}$$

We apply shifted QR iterations to this truncated factorization of $A$. Let $\mu$ be a shift and

$$H_{k+p} - \mu I = QR$$

be the QR factorization with $Q$ orthogonal and $R$ upper triangular. Then

$$H_{k+p} \otimes I_s - \mu I = (H_{k+p} - \mu I) \otimes I_s = (QR) \otimes I_s = (Q \otimes I_s)(R \otimes I_s)$$

and

$$(RQ)\otimes I_s+\mu I=(RQ+\mu I)\otimes I_s=(Q^HH_{k+p}Q)\otimes I_s.$$

Therefore, we get

$$\begin{aligned}
&(A-\mu I)\mathcal{V}_{k+p}-\mathcal{V}_{k+p}(H_{k+p}\otimes I_s-\mu I)=r_{k+p}(e_{k+p}^H\otimes I_s),\\
&(A-\mu I)\mathcal{V}_{k+p}-\mathcal{V}_{k+p}(Q\otimes I_s)(R\otimes I_s)=r_{k+p}(e_{k+p}^H\otimes I_s),\\
&(A-\mu I)\big(\mathcal{V}_{k+p}(Q\otimes I_s)\big)-\big(\mathcal{V}_{k+p}(Q\otimes I_s)\big)\big((RQ)\otimes I_s\big)=r_{k+p}\big((e_{k+p}^HQ)\otimes I_s\big),\\
&A\big(\mathcal{V}_{k+p}(Q\otimes I_s)\big)-\big(\mathcal{V}_{k+p}(Q\otimes I_s)\big)\big((RQ)\otimes I_s+\mu I\big)=r_{k+p}\big((e_{k+p}^HQ)\otimes I_s\big),
\end{aligned}$$

i.e.,

$$A\mathcal{V}_{k+p}(Q\otimes I_s)=\big(\mathcal{V}_{k+p}(Q\otimes I_s),V_{k+p+1}\big)\begin{pmatrix}Q^HH_{k+p}Q\\ \beta_{k+p}e_{k+p}^HQ\end{pmatrix}\otimes I_s. \tag{35}$$

A successive application of $p$ implicit shifts results in

$$A\mathcal{V}_{k+p}^+=(\mathcal{V}_{k+p}^+,V_{k+p+1})\begin{pmatrix}H_{k+p}^+\\ \beta_{k+p}e_{k+p}^H\hat{Q}\end{pmatrix}\otimes I_s, \tag{36}$$

where $\mathcal{V}_{k+p}^+=\mathcal{V}_{k+p}(\hat{Q}\otimes I_s)$, $H_{k+p}^+=\hat{Q}^HH_{k+p}\hat{Q}$ and $\hat{Q}=Q_1Q_2\ldots Q_p$, with $Q_j$ the orthogonal matrix associated with the shift $\mu_j$. Now, partition

$$\mathcal{V}_{k+p}^+=(\mathcal{V}_k^+,\hat{\mathcal{V}}_k),\qquad H_{k+p}^+=\begin{pmatrix}H_k^+ & M\\ \hat{\beta}_ke_1e_k^H & \hat{H}_p\end{pmatrix}$$

and note

$$\beta_{k+p}e_{k+p}^H\hat{Q}=(0,0,\ldots,\tilde{\beta}_{k+p},b^H).$$

So we get

$$A(\mathcal{V}_k^+,\hat{\mathcal{V}}_p)=(\mathcal{V}_k^+,\hat{\mathcal{V}}_p,V_{k+p+1})\begin{pmatrix}H_k^+ & M\\ \hat{\beta}_ke_1e_k^H & \hat{H}\\ \tilde{\beta}_{k+p}e_k^H & b^T\end{pmatrix}\otimes I_s. \tag{37}$$

Equating the first $k$ columns on both sides of (37) gives

$$A\mathcal{V}_k^+=\mathcal{V}_k^+(H_k^+\otimes I_s)+r_k^+(e_k^H\otimes I_s), \tag{38}$$

where $V_{k+1}^+=(1/\beta_k^+)r_k^+$, $r_k^+\equiv\hat{\mathcal{V}}_p(e_1\otimes I_s)\hat{\beta}_k+V_{k+p+1}\tilde{\beta}_{k+p}$ and $\beta_k^+=\|r_k^+\|$. Note that

$$tr\big((\mathcal{V}_k^+)^H\hat{\mathcal{V}}_p(e_1\otimes I_s)\big)=0$$

and

$$tr\big((\mathcal{V}_k^+)^HV_{k+p+1}\big)=0.$$

So $tr\big((\mathcal{V}_k^+)^H V_{k+1}\big) = 0$. This is a new $k$-step global Arnoldi process starting with the updated $\mathcal{V}_1^+$, so we do not need to restart from scratch and extend it to an $m$-step one from step $k+1$ upwards in a standard way.

Similarly to the implicitly restarted Arnoldi algorithm (IRA) [33], we can use those $p$ unwanted F-Ritz values as shifts, also called the exact shifts. So we have developed an implicitly restarted global Arnoldi algorithm (IRGA) with the exact shifts suggested.

**Algorithm 4.** IRGA with the exact shifts

1. Given the number $k$ of desired eigenpairs, choose $s, m$ and let $p = m - k$, and take $V_1 = V/\|V\|_F$ as an $n \times s$ starting matrix.
2. Run the $m$-step global Arnoldi process to get $[H, \mathcal{V}, k]$.
3. Compute the eigenpairs of $H$, select $k$ eigenvalues of $H$ as approximations to the desired eigenvalues and take the $p$ unwanted eigenvalues as shifts.
4. Apply implicit restarting approach with $p$ shifts, and the update the global Arnoldi algorithm $V \leftarrow VQ$, $H \leftarrow Q^T HQ$ is returned.
5. Test convergence. If yes, stop; otherwise, go to Step 2 and extend the global Arnoldi process from step $k+1$ upwards.

In order to determine the multiplicities of the desired eigenvalues, we combine Algorithm 3 with Algorithms 4 and present the following algorithm.

**Algorithm 5.** IRGA for multiple eigenproblems

1. Given the number $k$ of desired eigenpairs, choose $s, m$ and let $p = m - k$. Define the set $\mathcal{S} = \{1, 2, \ldots, k\}$, $\Phi = \emptyset$, $q = 1$ and prescribe a tolerance tol.
2. Take $V_1 = V/\|V\|_F$ as an $n \times s_q$ starting matrix.
3. Run the $m$-step global Arnoldi process to get $[H, \mathcal{V}, k]$.
4. Compute the eigenpairs $(\lambda_i^{(m)}, y_i^{(m)})$, $i = 1, 2, \ldots, m$ of $H$, select $\lambda_i^{(m)}$, $i = 1, 2, \ldots, k$ as approximations to the desired eigenvalues $\lambda_i$, $i = 1, 2, \ldots, k$ and take the $p$ unwanted $\lambda_i^{(m)}$, $i = k+1, \ldots, m$ as shifts.
5. Apply implicit restarting approach with $p$ shifts, and the update the global Arnoldi algorithm $V \leftarrow VQ$, $H \leftarrow Q^T HQ$ is returned.
6. Test convergence. If yes, go to Step 7; otherwise, go to Step 3 and extend the global Arnoldi process from step $k+1$ upwards.
7. For all $i \in \mathcal{S}$, set $\Phi_{is}^{(m)} = (\Psi, \varphi_{i1}^{(m)}, \varphi_{i2}^{(m)}, \ldots, \varphi_{is_q}^{(m)})$ and $s$ the number of its columns.
   (a) Compute the numerical rank $r_s$ of $\Phi_{is}^{(m)}$ for all $i \in \mathcal{S}$;
   (b) If $r_s < s$, set $d_i = r_s$ and remove $i$ from $\mathcal{S}$;
   (c) Otherwise, let $\Psi = \Phi_{is}^{(m)}$, $q = q + 1$ and go to Step 2.

We comment that if $A$ is real symmetric then Algorithm 4 naturally works by simplifying the global Arnoldi process as a symmetric global Lanczos process. For $s = 1$, Algorithm 4 mathematically reduces to the implicitly restarted Arnoldi algorithm [33], whose standard Matlab code is the function eigs.m.

## 6. Numerical Experiments

We present numerical examples to illustrate the efficiency and reliability of IRGA. All experiments were run on a PC with 2.2 GHz Intel Core 2 Duo T7500 processor using MATLAB 7.1 with the machine precision $\epsilon = 2.22 \times 10^{-16}$ under the Window XP operating system. If the relative residual norm

$$\frac{\|r_{ij}^{(m)}\|}{\|A\|_1} = \frac{\|(A - \lambda_i^{(m)} I)\varphi_{ij}^{(m)}\|}{\|A\|_1} \leq tol$$

with $tol$ a prescribed tolerance, then $(\lambda_i^{(m)}, \varphi_{ij}^{(m)})$ is accepted to have converged. Based on the previous analysis, we assume that

$$C_{ij} \leq C'_{ij} = \frac{10^{-3}}{\max_{1 \leq j \leq s} \|r_{ij}^{(m)}\|}$$

in (32), which allows the multiple eigenproblems to be quite ill conditioned for small $tol$. As $tol$ diminishes, Algorithm 5 is more applicable and works for worse conditioned multiple eigenproblems.

We took random $V_1$'s in all the examples. In all the tables below, we denote by $iter$ the number of restarts and by $Residual\ norms$ the above relative residual norms.

**Example** 1. This test problem is taken from the set of test matrices in $netlib$, and the matrix $A$ is of form

$$\begin{pmatrix} 0 & 1 & & & \\ n-1 & 0 & 2 & & \\ & \ddots & \ddots & \ddots & \\ & & 2 & 0 & n-1 \\ & & & 1 & 0 \end{pmatrix}$$

$A$ has zero diagonal entries and known eigenvalues, and is singular if $n$ is odd. The eigenvalues are plus and minus $n-1$, $n-3$, $n-5, \ldots$, (1 or 0). The eigenvalue problem of $A$ is highly ill conditioned for $n = 2000$, the computed condition number of the eigenvector matrix is $4.5e+220$. Moreover, the smallest spectral condition number of an individual eigenvalue is large up to $5.6e+201$. So it is expected that it may be very difficult to compute a few largest eigenpairs of $A$ using Arnoldi type algorithms.

We perform Algorithm 4 on the $2000 \times 2000$ matrix $A$, and require that the algorithm stops as soon as all actual residual norms of the approximating eigenpairs are below $tol = 10^{-6}$. We want to compute the four largest eigenvalues $1999, 1997, 1995, 1993$. Table 1 shows the results obtained. The left part of Figure 1 depicts the convergence curves for $m = 30$. We see the algorithm used almost the same restarts to achieve the prescribed accuracy for different $s$. This justifies that the global Arnoldi algorithm has the same convergence speed as the standard Arnoldi algorithm, i.e., $s = 1$, as far as $iter$ is concerned. It is also seen from the table that the bigger $m$ is, fewer restarts the algorithm uses and the algorithm converges faster for exterior eigenvalues.

Table 1. Example 1

| *m* | *s* | *iter* | Residual norms | | | |
|---|---|---|---|---|---|---|
| | | | 1 | 2 | 3 | 4 |
| 20 | 1 | 215 | 2.2e-13 | 1.0e-11 | 2.4e-9 | 4.9e-7 |
| | 2 | 206 | 3.2e-13 | 1.5e-11 | 3.3e-9 | 6.7e-7 |
| | 3 | 211 | 1.7e-13 | 1.1e-11 | 2.7e-9 | 6.2e-7 |
| 30 | 1 | 121 | 2.1e-13 | 6.5e-12 | 1.5e-9 | 3.6e-7 |
| | 2 | 124 | 2.7e-13 | 5.5e-12 | 1.4e-9 | 3.4e-7 |
| | 3 | 122 | 1.2e-13 | 3.0e-12 | 7.9e-10 | 2.1e-7 |
| 40 | 1 | 87 | 1.7e-13 | 3.4e-12 | 8.8e-10 | 2.4e-7 |
| | 2 | 87 | 1.6e-13 | 2.1e-12 | 5.2e-10 | 1.4e-7 |
| | 3 | 87 | 1.7e-13 | 2.7e-12 | 7.7e-10 | 2.4e-7 |

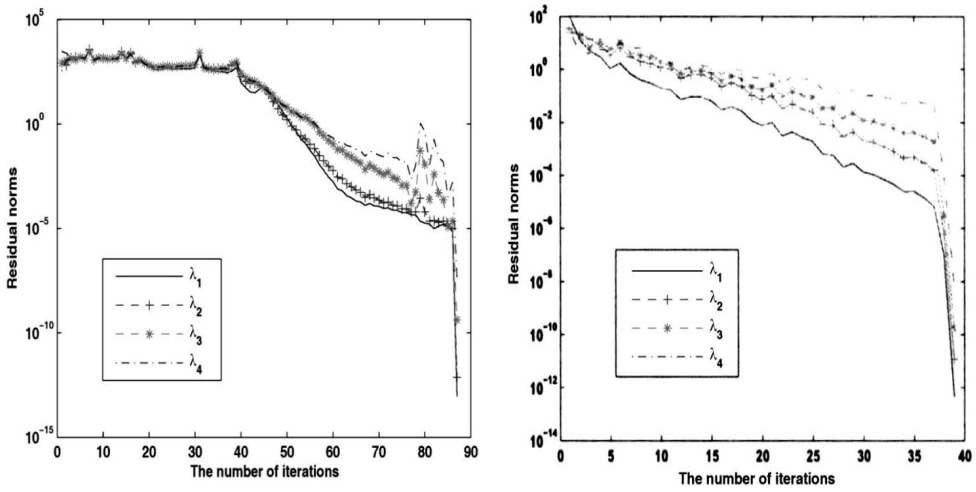


Fig. 1. Left: $A(2000)$ with $m = 30$, $s = 2$; right: dw8192 with $m = 20$, $s = 2$.

**Example** 2. We test the $8192 \times 8192$ unsymmetric matrix DW8192 from [3]. The stopping criterion and the notation used are as before. We compute the largest four eigenvalues in magnitude

$$\lambda_1 \approx -109.8278, \lambda_2 \approx -109.8236,$$
$$\lambda_3 \approx -109.8168, \lambda_4 \approx -109.8079,$$

which are quite clustered but numerically distinct. So this problem may not be easy for Arnoldi type algorithms. Table 2 reports the results, where we see the proposed algorithm solved the problem effectively and reliably. The right part of Figure 1 depicts the convergence curves for $m = 20$. Besides, we have observations similar to those for Example 1.

Table 2. Example 2

| *m* | *s* | *iter* | Residual norms | | | |
|---|---|---|---|---|---|---|
| | | | 1 | 2 | 3 | 4 |
| 15 | 1 | 114 | 2.7e-11 | 8.2e-10 | 1.4e-8 | 8.7e-7 |
| | 2 | 121 | 1.9e-12 | 5.9e-11 | 1.1e-9 | 6.9e-8 |
| | 3 | 122 | 1.8e-10 | 5.9e-10 | 3.2e-9 | 1.9e-7 |
| 20 | 1 | 39 | 7.2e-11 | 1.6e-11 | 2.2e-10 | 1.3e-8 |
| | 2 | 39 | 4.7e-13 | 1.2e-11 | 1.9e-10 | 1.2e-8 |
| | 3 | 31 | 3.7e-11 | 8.7e-10 | 1.1e-8 | 6.5e-7 |
| 25 | 1 | 21 | 7.2e-11 | 1.4e-9 | 1.3e-8 | 4.5e-7 |
| | 2 | 21 | 7.5e-11 | 1.4e-9 | 1.2e-8 | 3.8e-7 |
| | 3 | 25 | 1.1e-11 | 2.5e-10 | 3.1e-9 | 1.6e-7 |

**Example** 3. We test the matrix OLM1000 [3], a $1000 \times 1000$ real unsymmetric matrix. The stopping criterion as well as the notation used are as before. We want to compute the largest four eigenvalues in magnitude

$$\lambda_1 \approx -1.01633831, \lambda_2 \approx -1.01630831,$$
$$\lambda_3 \approx -1.01625831, \lambda_4 \approx -1.01618831,$$

which are quite clustered but numerically distinct. So like Example 1, this problem may not be easy for the proposed algorithm. Table 3 lists the results obtained. The right part of Figure 2 depicts the convergence curves for $m = 30$. From both the table and the figure, we see the algorithm solved the problem effectively and reliably. Again, we have similar observations to those for Example 1.

**Example** 4. Consider convection diffusion differential operator

$$-\triangle u(x, y) + \rho u_x(x, y) = \lambda u(x, y)$$

Table 3. Example 3

| *m* | *s* | *iter* | Residual norms | | | |
|---|---|---|---|---|---|---|
| | | | 1 | 2 | 3 | 4 |
| 20 | 1 | 169 | 1.0e-12 | 2.6e-11 | 4.3e-9 | 1.8e-7 |
| | 2 | 173 | 1.1e-12 | 2.9e-11 | 4.9e-9 | 2.2e-7 |
| | 3 | 169 | 7.3e-13 | 1.8e-11 | 2.9e-9 | 1.2e-7 |
| 30 | 1 | 73 | 3.2e-13 | 7.5e-12 | 1.2e-9 | 6.1e-8 |
| | 2 | 71 | 4.9e-13 | 1.1e-11 | 1.6e-9 | 6.7e-7 |
| | 3 | 71 | 2.9e-13 | 6.7e-12 | 1.1e-9 | 4.9e-8 |
| 40 | 1 | 43 | 1.1e-13 | 2.4e-12 | 3.4e-10 | 1.6e-8 |
| | 2 | 43 | 5.3e-12 | 1.1e-10 | 1.6e-8 | 7.5e-7 |
| | 3 | 43 | 5.6e-12 | 1.2e-10 | 1.7e-8 | 7.4e-7 |

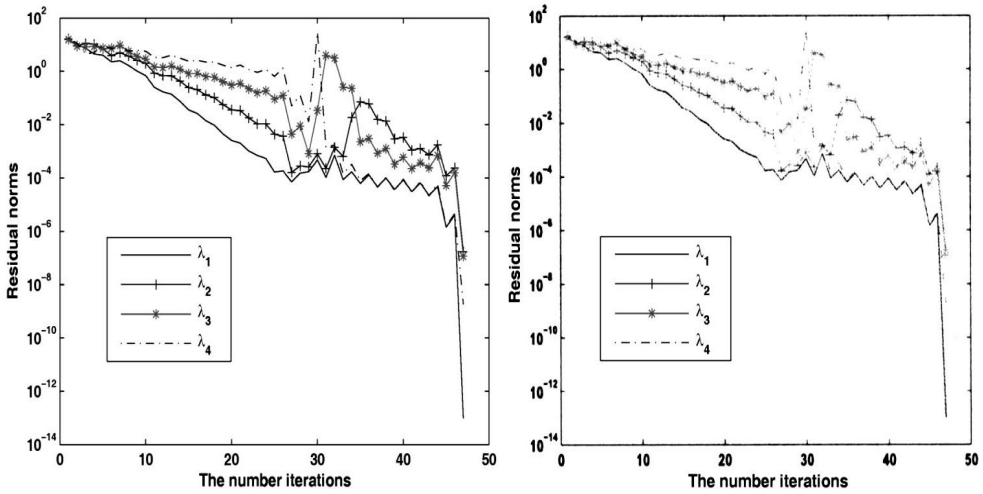


Fig. 2. Left: olm1000 with $m = 30$, $s = 2$; right: $A_{100}$ with $m = 20$, $s = 2$.

on a square region $[0, 1] \times [0, 1]$ with the boundary condition $u(x, y) = 0$. Taking $\rho = 1$ and discretizing with centered differences yield a block tridiagonal matrix $A_n = tri(-I, B_n, -I)$, where $B_n = tri(b, 4, a)$ is a tridiagonal matrix with $a = -1 + 1/2(n+1)$ and $b = -1 - 1/2(n+1)$, and $n$ is chosen the number of interior mesh points on each side of the square. $A_n$ is of order $N = n^2$. The eigenvalues $\lambda_2$ and $\lambda_3$ are very clustered and they get closer as $n$ increases.

We test Algorithm 4 on the $10000 \times 10000$ matrix $A_{100}$ obtained by taking $n = 100$. We are interested in the four eigenvalues with the largest real parts, and the stopping criterion as well as the notation used are as before. The initial $V_1$ were

chosen randomly in a normal distribution, and the computed eigenvalues are

$$\lambda_1 \approx 7.99804063, \lambda_2 \approx 7.99513930$$
$$\lambda_3 \approx 7.99513926, \lambda_4 \approx 7.99223793$$

Table 4 lists the results obtained and the right part of Figure 2 depicts the convergence curves for $m = 20$.

Table 4. Example 4

| *m* | *s* | *iter* | Residual norms | | | |
|---|---|---|---|---|---|---|
| | | | 1 | 2 | 3 | 4 |
| 15 | 1 | 92 | 1.1e-13 | 7.5e-7 | 5.1e-7 | 1.5e-8 |
| | 2 | 97 | 9.3e-14 | 1.0e-7 | 6.9e-8 | 2.8e-9 |
| | 3 | 96 | 2.3e-13 | 4.9e-7 | 3.4e-7 | 8.9e-9 |
| 20 | 1 | 48 | 1.4e-13 | 4.4e-7 | 3.1e-7 | 5.5e-9 |
| | 2 | 47 | 1.3e-13 | 1.8e-7 | 1.2e-7 | 1.9e-9 |
| | 3 | 49 | 1.2e-13 | 2.2e-7 | 1.5e-7 | 3.3e-9 |
| 25 | 1 | 32 | 3.1e-13 | 7.3e-7 | 5.0e-7 | 6.7e-9 |
| | 2 | 33 | 6.4e-13 | 2.2e-7 | 1.5e-7 | 2.2e-9 |
| | 3 | 36 | 2.2e-13 | 5.1e-8 | 2.5e-8 | 8.4e-10 |

In what follows, we show how to use the global Arnoldi method and Algorithm 5 to solve multiple eigenproblems and determine the multiplicities $d_i$ of $\lambda_i$, $i = 1, 2, \ldots, k$ and the associated eigenspaces. Differently from that done in Examples 1–4, for each F-Ritz value we now compute all the $s$ F-Ritz vectors by (19) simultaneously. Examples 5–8 reports the numerical results, where in the tables we list all the residual norms of the $s$ F-Ritz pairs for each F-Ritz value used to approximate a desired eigenvalue.

**Example** 5. In this example, we test a multiple eigenproblem. Let $B = I_2 \otimes A$, where $A$ is the one presented in Example 1. The $4000 \times 4000$ real unsymmetric matrix $B$ has eigenvalues with multiplicity two. Algorithm 5 is run on $B$. As have been seen from Example 1, the eigenvalue problem of $A$ is very highly ill conditioned and $C_{ij}$ is very huge. We use the same stopping criterion and the notation as before. Both the number of restarts and CPU time (in seconds) are used to measure the cost of the algorithm. Table 5 reports the results for various $s$ and $m$ and the experiments of determining $d_i$ for $m = 30, 40$, where $svd(X)$ is the set of all the singular values of the matrix $X$ and $\Phi_{is}^{(m)} = (\Phi_{s_1}^{(m)}, \ldots, \Phi_{s_q}^{(m)})$ in all the tables. We see from Table 5 that Algorithm 5 has found $\lambda_i, d_i, i = 1, 2, 3, 4$ reliably. This is very surprising since the true $C_{ij}$ are much larger than $\frac{1}{tol}$. It is observed that for the same $m$ and different $s$ the algorithm used almost the same restarts and the bigger $s$ is, the more costly the algorithm is. However, if one is required to

determine $d_i$ and compute a basis of $\mathcal{P}_i$, then IRGA for $s > 1$ is preferable and advantageous to IRGA for $s = 1$, i.e., IRA, since it uses less CPU time. Note that for the same $m$, running IRGA for $s > 1$ once is less costly than running IRA $s$ times. For example, we have to run IRGA for $s = 1$ three times to achieve the aim but only need to run IRGA for each $s = 1, 2$ once or IRGA for $s = 3$ once, while the latter is cheaper than the former.

Table 5. Example 5

| $m$ | $s$ | *iter* | *cpu time* | Residual norms | | | |
|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 |
| 20 | 1 | 221 | 50.2 | 2.7e-13 | 1.6e-12 | 1.5e-9 | 4.2e-7 |
| | 2 | 209 | 82.8 | 6.0e-13 | 2.7e-11 | 1.2e-8 | 8.2e-7 |
| | | | | 2.8e-13 | 6.7e-11 | 7.0e-9 | 9.5e-7 |
| | 3 | 221 | 123.5 | 4.2e-13 | 1.2e-11 | 7.4e-9 | 6.9e-7 |
| | | | | 2.5e-13 | 2.7e-11 | 3.8e-9 | 7.3e-7 |
| | | | | 2.5e-13 | 1.1e-11 | 2.6e-9 | 4.6e-7 |
| 30 | 1 | 127 | 63.6 | 3.8e-13 | 1.6e-12 | 1.4e-9 | 3.4e-7 |
| | 2 | 123 | 116.9 | 3.2e-13 | 6.7e-12 | 3.5e-9 | 2.9e-7 |
| | | | | 1.8e-13 | 1.5e-11 | 1.8e-9 | 2.9e-7 |
| | 3 | 123 | 176.3 | 5.1e-13 | 4.6e-12 | 2.5e-9 | 2.1e-7 |
| | | | | 3.2e-13 | 1.4e-11 | 1.7e-9 | 3.0e-7 |
| | | | | 3.4e-13 | 6.6e-12 | 1.4e-9 | 2.1e-8 |
| 40 | 1 | 87 | 47.7 | 2.6e-13 | 1.2e-11 | 7.0e-9 | 7.2e-7 |
| | 2 | 89 | 90.7 | 3.4e-13 | 2.3e-12 | 1.4e-9 | 1.4e-7 |
| | | | | 2.1e-13 | 4.9e-12 | 6.6e-10 | 1.4e-7 |
| | 3 | 89 | 123.8 | 5.8e-13 | 6.5e-12 | 3.4e-9 | 2.9e-7 |
| | | | | 2.9e-13 | 1.6e-11 | 1.9e-9 | 3.4e-7 |
| | | | | 2.7e-13 | 7.2e-12 | 1.4e-9 | 2.3e-7 |

| $m$ | $q$ | $s_q$ | svd($\Phi_{1s}^{(m)}$) | svd($\Phi_{2s}^{(m)}$) | svd($\Phi_{3s}^{(m)}$) | svd($\Phi_{4s}^{(m)}$) |
|---|---|---|---|---|---|---|
| 30 | 1 | $s_1 = 2$ | 1.32767555 | 1.15555006 | 1.13624505 | 1.01783812 |
| | | | 0.48711153 | 0.81529385 | 0.84199002 | 0.98183785 |
| | 2 | $s_2 = 1$ | 1.47590909 | 1.46385174 | 1.47963570 | 1.41339586 |
| | | | 0.90647248 | 0.92581752 | 0.90037670 | 1.00115540 |
| | | | 2.2e-12 | 1.6e-10 | 6.6e-9 | 7.0e-6 |
| 40 | 1 | $s_1 = 3$ | 1.66060661 | 1.33537867 | 1.48111915 | 1.41625684 |
| | | | 0.49232682 | 1.10307018 | 0.89793434 | 0.99710409 |
| | | | 6.7e-12 | 2.6e-11 | 5.9e-10 | 3.6e-6 |

| | $\lambda_1$ | $\lambda_2$ | $\lambda_3$ | $\lambda_4$ |
|---|---|---|---|---|
| Multiplicity | 2 | 2 | 2 | 2 |

**Example** 6. The matrix $B = A \otimes I_2$, where $A$ is the one presented in Example 2. This $16384 \times 16384$ real unsymmetric matrix $B$ has eigenvalues with multiplicity

two, and the four desired eigenvalues are clustered. We find $g_1 \approx g_2 \approx 4.2 \times 10^{-3}$ and $g_3 \approx g_4 \approx 6.9 \times 10^{-3}$. So $C_{1j} \geq \frac{1}{g_1} \approx 238.1$, $C_{2j} \geq \frac{1}{g_2} \approx 238.1$, $C_{3j} \geq \frac{1}{g_3} \approx 144.9$, $C_{4j} \geq \frac{1}{g_4} \approx 144.9$. The eigenvectors associated with $\lambda_1, \ldots, \lambda_4$ may be moderately ill conditioned. Algorithm 5 is run on $B$. The stopping criterion and the notation used are as before. Table 6 lists the results. We see from Table 6 that Algorithm 5 has found $d_i, i = 1, 2, 3, 4$ reliably. Other observations are similar to those for Example 5.

Table 6. Example 6

| *m* | *s* | *iter* | *cpu time* | Residual norms | | | |
|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 |
| 15 | 1 | 121 | 40.5 | 3.9e-12 | 1.3e-10 | 2.2e-8 | 9.2e-8 |
| | 2 | 121 | 76.2 | 9.7e-12 | 3.1e-10 | 5.3e-9 | 2.1e-7 |
| | | | | 1.0e-11 | 2.6e-10 | 1.7e-8 | 8.5e-8 |
| | 3 | 124 | 112.6 | 4.6e-12 | 1.5e-12 | 2.6e-9 | 1.1e-7 |
| | | | | 4.6e-12 | 1.2e-10 | 8.3e-9 | 4.2e-8 |
| | | | | 5.8e-12 | 6.9e-11 | 3.5e-9 | 1.9e-7 |
| 20 | 1 | 39 | 27.3 | 8.6e-13 | 2.3e-11 | 3.5e-10 | 1.5e-8 |
| | 2 | 35 | 43.0 | 2.1e-13 | 5.1e-12 | 7.4e-11 | 3.3e-9 |
| | | | | 2.3e-13 | 4.7e-12 | 2.7e-10 | 1.5e-9 |
| | 3 | 38 | 71.9 | 3.7e-11 | 8.7e-10 | 1.1e-8 | 6.5e-7 |
| | | | | 5.5e-13 | 1.4e-11 | 2.2e-10 | 9.6e-9 |
| | | | | 6.8e-13 | 6.8e-12 | 2.9e-10 | 1.8e-8 |
| 25 | 1 | 21 | 22.5 | 1.4e-10 | 2.7e-9 | 2.3e-8 | 4.7e-7 |
| | 2 | 23 | 43.4 | 2.5e-11 | 1.2e-9 | 1.3e-8 | 3.7e-7 |
| | | | | 5.9e-11 | 1.0e-9 | 4.3e-8 | 1.5e-7 |
| | 3 | 21 | 62.9 | 1.0e-10 | 2.1e-9 | 2.0e-8 | 5.4e-7 |
| | | | | 1.1e-11 | 2.5e-10 | 3.1e-9 | 1.6e-7 |
| | | | | 1.2e-10 | 9.6e-10 | 2.7e-8 | 9.8e-7 |

| *m* | *q* | $s_q$ | svd($\Phi_{1s}^{(m)}$) | svd($\Phi_{2s}^{(m)}$) | svd($\Phi_{3s}^{(m)}$) | svd($\Phi_{4s}^{(m)}$) |
|---|---|---|---|---|---|---|
| 20 | 1 | $s_1 = 2$ | 1.38171595 | 1.41311336 | 1.32751084 | 1.40289147 |
| | | | 0.30143164 | 0.05577303 | 0.48756023 | 0.17859316 |
| | 2 | $s_2 = 1$ | 1.69691398 | 1.73085332 | 1.63961645 | 1.71975292 |
| | | | 0.34710652 | 0.64395568 | 0.55826330 | 0.20603350 |
| | | | 1.9e-11 | 2.9e-11 | 2.4e-10 | 1.3e-8 |
| 26 | 1 | $s_1 = 3$ | 1.63011262 | 1.51812026 | 1.56652967 | 1.55133022 |
| | | | 0.58543391 | 0.83385303 | 0.73890784 | 0.77030808 |
| | | | 2.3e-9 | 2.9e-8 | 5.9e-7 | 2.1e-5 |

| | $\lambda_1$ | $\lambda_2$ | $\lambda_3$ | $\lambda_4$ |
|---|---|---|---|---|
| Multiplicity | 2 | 2 | 2 | 2 |

**Example** 7. We construct a $2000 \times 2000$ matrix $B = A \otimes I_2$, where $A$ is the one presented in Example 3. The matrix $B$ has eigenvalues with multiplicity two

and the four desired eigenvalues are clustered. We find $g_1 \approx g_2 \approx 3.0 \times 10^{-5}$ and $g_3 \approx g_4 \approx 5.0 \times 10^{-5}$, so $C_{1j} \geq \frac{1}{g_1} \approx 33333$, $C_{2j} \geq \frac{1}{g_2} \approx 33333$, $C_{3j} \geq \frac{1}{g_3} \approx 20000$, $C_{4j} \geq \frac{1}{g_4} \approx 20000$. It is clear that the eigenvectors associated with $\lambda_1, \ldots, \lambda_4$ are quite ill-conditioned Algorithm 5 is run on $B$. The stopping criterion and the notation used are as before. Table 7 reports the results. We see from Table 7 that for this ill-conditioned problem Algorithm 5 has found $d_i, i = 1, 2, 3, 4$ reliably. Other observations are similar to those for Example 5.

Table 7. Example 7

| $m$ | $s$ | *iter* | *cpu time* | Residual norms | | | |
|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 |
| 20 | 1 | 173 | 19.2 | 2.3e-12 | 5.4e-11 | 4.3e-9 | 4.5e-7 |
| | 2 | 169 | 24.8 | 2.6e-12 | 5.9e-11 | 4.2e-9 | 3.8e-7 |
| | | | | 2.4e-12 | 2.5e-10 | 3.4e-9 | 1.5e-7 |
| | 3 | 169 | 42.1 | 3.4e-12 | 7.3e-11 | 5.0e-9 | 4.3e-7 |
| | | | | 2.9e-12 | 2.9e-10 | 3.8e-9 | 1.6e-7 |
| | | | | 1.7e-12 | 1.3e-10 | 1.9e-9 | 2.7e-7 |
| 30 | 1 | 71 | 16.2 | 9.9e-13 | 2.1e-11 | 1.4e-9 | 1.4e-7 |
| | 2 | 73 | 30.6 | 8.0e-13 | 1.7e-11 | 1.2e-9 | 1.3e-7 |
| | | | | 6.3e-13 | 6.1e-11 | 8.2e-10 | 4.1e-8 |
| | 3 | 73 | 45.9 | 1.1e-12 | 2.2e-11 | 1.5e-9 | 1.5e-7 |
| | | | | 9.1e-13 | 8.7e-11 | 1.2e-9 | 5.3e-8 |
| | | | | 5.6e-13 | 4.1e-11 | 6.0e-10 | 9.4e-8 |
| 40 | 1 | 44 | 18.4 | 2.1e-13 | 4.3e-12 | 2.9e-10 | 3.5e-10 |
| | 2 | 43 | 27.9 | 9.8e-13 | 1.9e-11 | 1.1e-9 | 1.1e-7 |
| | | | | 8.7e-13 | 7.5e-11 | 8.7e-10 | 4.1e-8 |
| | 3 | 43 | 44.0 | 8.5e-13 | 1.6e-11 | 9.7e-10 | 9.1e-8 |
| | | | | 7.2e-13 | 6.2e-11 | 7.1e-10 | 3.3e-8 |
| | | | | 4.2e-13 | 2.8e-11 | 3.7e-10 | 5.5e-8 |

| $m$ | $q$ | $s_q$ | svd($\Phi_{1s}^{(m)}$) | svd($\Phi_{2s}^{(m)}$) | svd($\Phi_{3s}^{(m)}$) | svd($\Phi_{4s}^{(m)}$) |
|---|---|---|---|---|---|---|
| 30 | 1 | $s_1 = 2$ | 1.08552960 | 1.40845652 | 1.39162800 | 1.40856809 |
| | | | 0.90643560 | 0.12747635 | 0.25173699 | 0.12623762 |
| | 2 | $s_2 = 1$ | 1.44198899 | 1.42070899 | 1.70214617 | 1.72399290 |
| | | | 0.95951433 | 0.99075021 | 0.32046595 | 0.16687869 |
| | | | 6.7e-11 | 5.8e-10 | 3.9e-8 | 4.1e-6 |
| 40 | 1 | $s_1 = 3$ | 1.37439500 | 1.71477436 | 1.58708408 | 1.47576677 |
| | | | 1.05405805 | 0.24402641 | 0.69365994 | 0.90670416 |
| | | | 3.2e-11 | 2.2e-9 | 4.9e-8 | 3.6e-6 |

| | $\lambda_1$ | $\lambda_2$ | $\lambda_3$ | $\lambda_4$ |
|---|---|---|---|---|
| Multiplicity | 2 | 2 | 2 | 2 |

**Example** 8. This test matrix $B = A \otimes I_2$ is a $20000 \times 20000$ matrix, where $A$

is the one presented in Example 4. The matrix $B$ has eigenvalues with multiplicity two and the four desired eigenvalues are clustered. We find $g_1 \approx 2.9 \times 10^{-3}$, $g_2 \approx g_3 \approx 4.0 \times 10^{-7}$ and $g_4 \approx 2.9 \times 10^{-3}$, so $C_{1j} \geq \frac{1}{g_1} \approx 344.8$, $C_{2j} \geq \frac{1}{g_2} \approx 2.5 \times 10^6$, $C_{3j} \geq \frac{1}{g_3} \approx 2.5 \times 10^6$, $C_{4j} \geq \frac{1}{g_4} \approx 344.8$. It is seen that the eigenvectors associated with $\lambda_1, \lambda_4$ may be moderately ill conditioned but the ones with $\lambda_2, \lambda_3$ are definitely very ill conditioned. Algorithm 5 is run on $B$. The stopping criterion and the notation used are as before. Table 8 lists the results. We see from Table 8 that for this very ill-conditioned problem Algorithm 5 has found $d_i, i = 1, 2, 3, 4$ reliably. Other observations are similar to those for Example 5.

Table 8. Example 8

| *m* | *s* | *iter* | *cpu time* | Residual norms | | | |
|---|---|---|---|---|---|---|---|
| | | | | 1 | 2 | 3 | 4 |
| 20 | 1 | 97 | 36.7 | 2.6e-13 | 2.0e-7 | 1.4e-7 | 5.5e-9 |
| | 2 | 81 | 63.3 | 6.4e-13 | 7.4e-7 | 4.9e-7 | 6.1e-9 |
| | | | | 6.3e-13 | 2.9e-7 | 8.8e-7 | 1.3e-8 |
| | 3 | 82 | 107.6 | 3.7e-13 | 9.6e-7 | 6.2e-7 | 8.7e-9 |
| | | | | 3.7e-13 | 4.3e-7 | 1.3e-6 | 2.1e-8 |
| | | | | 3.3e-13 | 1.1e-6 | 7.3e-7 | 1.1e-8 |
| 30 | 1 | 47 | 36.8 | 2.6e-13 | 3.2e-7 | 2.2e-7 | 3.3e-9 |
| | 2 | 48 | 68.9 | 5.3e-13 | 2.6e-7 | 1.8e-7 | 2.9e-9 |
| | | | | 5.5e-13 | 1.2e-7 | 3.5e-7 | 6.9e-9 |
| | 3 | 49 | 110.5 | 6.3e-13 | 5.9e-7 | 6.2e-7 | 8.7e-9 |
| | | | | 6.1e-13 | 2.6e-7 | 7.6e-7 | 1.9e-8 |
| | | | | 5.3e-13 | 6.9e-7 | 4.8e-7 | 1.1e-8 |
| 40 | 1 | 33 | 37.8 | 1.2e-12 | 4.2e-7 | 2.8e-7 | 4.1e-9 |
| | 2 | 36 | 64.5 | 1.9e-13 | 1.7e-7 | 1.1e-7 | 2.4e-9 |
| | | | | 3.4e-13 | 7.3e-8 | 2.2e-7 | 5.7e-9 |
| | 3 | 34 | 108.7 | 6.3e-13 | 4.5e-7 | 2.9e-7 | 4.4e-9 |
| | | | | 6.2e-13 | 1.9e-7 | 5.5e-7 | 1.0e-8 |
| | | | | 5.3e-13 | 4.8e-7 | 3.3e-7 | 5.2e-9 |

| *m* | *q* | $s_q$ | svd($\Phi_{1s}^{(m)}$) | svd($\Phi_{2s}^{(m)}$) | svd($\Phi_{3s}^{(m)}$) | svd($\Phi_{4s}^{(m)}$) |
|---|---|---|---|---|---|---|
| 30 | 1 | $s_1 = 2$ | 1.41149692 | 1.03670228 | 1.32663736 | 1.18748200 |
| | | | 0.08761526 | 0.96189833 | 0.48993195 | 0.76804069 |
| | 2 | $s_2 = 1$ | 1.63061080 | 1.40278875 | 1.49036093 | 1.49447385 |
| | | | 0.58404488 | 1.01596441 | 0.88251021 | 0.87552722 |
| | | | 1.9e-12 | 1.8e-4 | 1.7e-4 | 2.0e-8 |
| 40 | 1 | $s_1 = 3$ | 1.66426522 | 1.43314186 | 1.39875005 | 1.54867003 |
| | | | 0.47981380 | 0.97267898 | 1.02151753 | 0.77564241 |
| | | | 5.8e-13 | 4.5e-5 | 4.9e-4 | 8.9e-9 |

| | $\lambda_1$ | $\lambda_2$ | $\lambda_3$ | $\lambda_4$ |
|---|---|---|---|---|
| Multiplicity | 2 | 2 | 2 | 2 |

## References


1. W. E. Arnoldi, The principle of minimized iteration in the solution of the matrix eigenvalue problem, *Quart. Appl. Math.*, **9** (1951), 11-29.
2. Z. Bai, J. Demmel, J. Dongarra, A. Ruhe and H. A. van der Vorst (eds.), *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, SIAM, Philadelphia, 2000.
3. Z. Bai, R. Barrett, D. Day, J. Demmel and J. Dongarra, *Test matrix collection (non-Hermitian eigenvalue problems)*, http://math.nist.gov/MarketMatrix.
4. L. Bao, Y. Lin and Y. Wei, A new projection method for solving large Sylvester equations, *Appl. Numer. Math.*, **57** (2007), 521-532.
5. M. Bellalij, K. Jbilou and H. Sadok, *New convergence results on the global GMRES method for diagonalizable matrices*, Technical Report, L. M. P. A., 2007.
6. F. Chatelin and D. Ho, Arnoldi-Tchebychev procedure for large scale nonsymmetric matrices, *Math. Model. Numer. Anal.*, **24** (1990), 53-65.
7. C. C. Chu, M. H. Lai and W. S. Feng, MIMO interconnects order reductions by using the multiple point adaptive-order rational global Arnoldi algorithm, *IEICE Trans. Elect.*, **E89-C** (2006), 792-808.
8. C. C. Chu, M. H. Lai and W. S. Feng, The multiple point global Lanczos method for mutiple-inputs multiple-outputs interconnect order reduction, *IEICE Trans. Elect.*, **E89-A** (2006), 2706-2716.
9. C. C. Chu, M. H. Lai and W. S. Feng, Model-order reductions for MIMO systems using global Krylov subspace methods, *Math. Comput. Simulat.*, to appear.
10. J. K. Cullum and W. E. Donath, A block Lanczos algorithm for computing the $q$ algebraically largest eigenvalues and a corresponding eigenspace for large, sparse symmetric matrices, in: *Proceedings of the 1994 IEEE Conference on Decision and Control*, IEEE Press, Piscataway, NJ, 1974, pp. 505-509.
11. J. W. Demmel, *Applied Numerical Linear Algebra*, SIAM, Philadelphia, 1997.
12. G. Golub and R. Underwood, The block Lanczos method for computing eigenvalues, in: *Mathematical Software III*, J. Rice (ed.), Academic Press, New York, 1977, pp. 364-377.
13. C. Gu and Z. Yang, Global SCD algorithm for real positive definite linear systems with multiple right-hand sides, *Appl. Math. Comput.*, **189** (2007), 59-67.
14. M. Heyouni, The global Hessenberg and CMRH methods for linear systems with multiple right-hand sides, *Numer. Algor.*, **26** (2001), 317-332.
15. M. Heyouni and K. Jbilou, Matrix Krylov subspace methods for large scale model reduction problems, *Appl. Math. Comput.*, **181** (2006), 1215-1228.

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

17. K. Jbilou, An Arnoldi based algorithm for large algebraic Riccati equations, *Appl. Math. Lett.*, **19** (2006), 437-444.

18. K. Jbilou, Low rank approximate solutions to large Sylvester matrix equations, *Appl. Math. Comput.*, **177** (2006), 365-376.

19. Z. Jia, The convergence of generalized Lanczos methods for large unsymmetric eigenproblems, *SIAM J. Matrix Anal. Appl.*, **16** (1995), 843-862.

20. Z. Jia, Generalized block Lanczos methods for large unsymmetric eigenproblems, *Numer. Math.*, **80** (1998), 239-266.

21. Z. Jia, Arnoldi type algorithms for large unsymmetric multiple eigenvalue problems, *J. Comput. Math.*, **17** (1999), 257-274.

22. W. Karush, An iterative method for finding characteristics vectors of a symmetric matrix, *Pacific J. Math.*, **1** (1951), 233-248.

23. R. B. Lehoucq and K. J. Maschhoff, *Implementation of an implicitly restarted block Arnoldi method*, Preprint MCS-P649-0297, Argonne National Laboratory, Argonne, IL, 1997.

24. Y. Lin, Implicitly restarted global FOM and GMRES for nonsymmetric matrix equations and Sylvester equations, *Appl. Math. Comput.*, **167** (2005), 1004-1025.

25. C. C. Paige, *The computation of eigenvalues and eigenvectors of very large sparse matrices*, Ph. D. thesis, London University, London, England, 1971.

26. Y. Saad, Variations on Arnoldi's method for computing eigenelements of large unsymmetric matrices, *Linear Algebra Appl.*, **34** (1980), 269-295.

27. Y. Saad, Projection methods for solving large sparse eigenvalue problems, in: *Matrix Pencils*, B. Kagström, A. Ruhe (eds.), Lecture Notes in Mathematics 973, Springer, Berlin, 1983, pp. 121-144.

28. Y. Saad, Chebyshev acceleration techniques for solving nonsymmetric eigenvalue problems, *Math. Comput.*, **42** (1984), 567-588.

29. Y. Saad, *Numerical Methods for Large Eigenvalue Problems*, Manchester University Press, 1992, UK.

30. D. K. Salkuyeh, CG-type algorithms to solve symmetric matrix equations, *Appl. Math. Comput.*, **172** (2006), 985-999.

31. D. K. Salkuyeh and F. Toutounian, New approaches for solving large Sylvester equations, *Appl. Math. Comput.*, **173** (2006), 9-18.

32. V. Simoncini and E. Gallopoulos, Convergence properties of block GMRES and matrix polynomials, *Linear Algebra Appl.*, **247** (1996), 97-119.

33. D. C. Sorensen, Implicit application of polynomial filters in a k-step Arnoldi method, *SIAM J. Matrix Anal. Appl.*, **13** (1992), 357-385.

34. G. W. Stewart, *Matrix Algorithms II: Eigensystems*, SIAM, Philadelphia, 2001.

Congying Duan and Zhongxiao Jia
Department of Mathematical Sciences
Tsinghua University
Beijing 100084
P. R. China
E-mail: duancy04@mails.thu.edu.cn
jiazx@tsinghua.edu.cn