A method for computing a few eigenpairs of large generalized eigenvalue problems

Maged Alkilayh[a], Lothar Reichel[b], Qiang Ye[c]

[a] *Department of Mathematical Sciences, Kent State University, Kent, OH 44242, USA, and Department of Mathematics, College of Science, Qassim University, Buraydah 51452, Saudi Arabia.*
[b] *Department of Mathematical Sciences, Kent State University, Kent, OH 44242, USA.*
[c] *Department of Mathematics, University of Kentucky, Lexington, KY 40506, USA.*

Abstract

Many methods for the computation of selected eigenpairs of generalized eigenproblems for matrix pairs use a shift-and-invert technique. When applied to large-scale problems, this requires the solution of large linear systems of equations. This paper proposes an application of an Arnoldi method described in [H. Voss, An Arnoldi method for nonlinear eigenvalue problems, BIT Numer. Math., 44 (2004), pp. 387–401] to the computation of a few extreme eigenpairs of a matrix pair. An advantage of this approach, when compared to methods that use the shift-and-invert technique, is that no large systems of equations have to be solved. We compare this approach to using a technique for simultaneously reducing a pair of large matrices to a pair of small matrices by a generalized Arnoldi process described in [R.-C. Li and Q. Ye, A Krylov subspace method for quadratic matrix polynomials with application to constrained least squares problems, SIAM J. Matrix Anal. Appl., 25 (2003), pp. 405–428] and [L. Hoffnung, R.-C. Li, and Q. Ye, Krylov type subspace methods for matrix polynomials, Linear Algebra Appl., 415 (2006), pp. 52–81]. The latter technique does not require the solution of large linear systems of equations either. Computed examples show the proposed method to yield approximations of the desired eigenpairs of higher accuracy when using about the same amount of computer storage space.

Keywords: generalized Krylov subspace method, generalized eigenvalue problem, large-scale problem
2020 MSC: 65F15

1. Introduction

This paper discusses an application of the Arnoldi method for the solution of nonlinear eigenvalue problems described by Voss [21] to the computation a few extreme eigenvalues and associated eigenvectors of large-scale generalized eigenproblems of

Email addresses: malkilay@kent.edu (Maged Alkilayh), reichel@math.kent.edu (Lothar Reichel), qye3@uky.edu (Qiang Ye)

the form

$$Ax = \lambda Bx, \qquad A, B \in \mathbb{C}^{n\times n}, \qquad x \in \mathbb{C}^n,\ \lambda \in \mathbb{C}, \tag{1.1}$$

where at least one of the matrices A or B are non-Hermitian. The need to determine a few eigenvalues and associated eigenvectors of large generalized eigenproblems of this kind arises in various applications including in structural engineering, where A is referred to as the stiffness matrix and B as the mass matrix, and in linear stability analysis of fluid flow. In the latter application the matrix A typically is nonsymmetric.

When the matrix B is Hermitian positive definite with a structure that allows its Cholesky factorization to be calculated for a reasonable cost, the problem (1.1) can be transformed to a standard eigenvalue problem; see, e.g., Baglama et al. [2] for computed illustrations for real matrices A and B. The most popular methods for computing a few selected eigenvalue-eigenvector pairs of (1.1) are based on the shift-and-invert technique, in which the generalized eigenproblem (1.1) is replaced by the standard eigenvalue problem

$$(A - \sigma B)^{-1} Bx = \mu x, \tag{1.2}$$

where the shift $\sigma \in \mathbb{C}$ is chosen to be close to the desired eigenvalues of the generalized eigenproblem (1.1), but not equal to an eigenvalue. The eigenvectors of (1.2) are eigenvectors of (1.1), and eigenvalues λ of (1.1) can be determined from eigenvalues μ of (1.2). The shift-and-invert technique requires the solution of one or several linear systems of equations with matrices of the form $A - \sigma B$.

The shift-and-invert technique (1.2) was first described by Ericsson and Ruhe [3] for the situation when both the matrices A and B are Hermitian, and has been generalized in various ways since, in particular to allow non-Hermitian matrices A and B, and to allow a Cayley transformation; see, e.g., Jia and Zhang [7], Saad [18, Chapter 9], and Sorensen [20]; Grimes et al. [5] require the matrix A to be symmetric and the matrix B to be symmetric positive semidefinite. The Jacobi-Davidson method described by Sleijpen et al. [19] is attractive when a suitable preconditioner is available; we do not require this in the present paper. Further discussions of methods for the computation of a few selected eigenpairs in the situation when both the matrices A and B are Hermitian are provided by Golub and Ye [4], Lampe and Voss [11], Pandur [14], Quillen and Ye [15], and Saad [18, Chapter 9].

We are concerned with the situation when the solution of one or several linear systems of equations with matrices of the form $A - \sigma B$, required when applying the transformation (1.2), is not possible or not attractive due to the large size or lack of suitable structure of the matrices A or B. There are not many methods available for determining a few eigenpairs of generalized eigenproblems (1.1) with large general square matrices A and B that do not require the solution of linear systems of equations with a large matrix. The solution of these systems may be quite time consuming if the matrices A and B do not have a structure that can be exploited, such as being diagonally dominant and banded. We are interested in methods that avoid the solution of large linear systems of equations. Note that the MATLAB function eigs, when applied to determine a few eigenpairs of the generalized eigenvalue problem (1.1), uses a shift-and-invert technique and therefore requires the solution of linear systems of equations with large matrices that are determined by A and B.

Hoffnung, Li, and Ye [6, 12] present a generalization of the Arnoldi method that simultaneously reduces a pair of large matrices to a pair of small matrices of generalized upper Hessenberg form. They apply this method to the solution of large-scale quadratic eigenvalue problems. We will describe an application to the reduction of the matrix pair $\{A,B\}$ to a pair of small matrices and use the latter to determine Ritz vectors of the matrix pair $\{A,B\}$ as approximations of eigenvectors of (1.1). Also, we will generalize some residual convergence bounds to this case.

It is the aim of the present paper to compare the generalized Arnoldi process by Hoffnung, Li, and Ye [6, 12] to the Arnoldi method proposed by Voss [21] for nonlinear eigenproblems when applied to the solution of generalized eigenvalue problems. The latter method constructs a solution subspace that is determined by residual vectors associated with available eigenpair approximations of (1.1). Neither approach requires the solution of large linear systems of equations.

This paper is organized as follows. Section 2 describes a restarted Arnoldi method inspired by the Arnoldi method proposed by Voss [21] to the computation of a few selected eigenpairs of (1.1). The application of the generalized Arnoldi method by Hoffnung, Li, and Ye [6, 12] is discussed in Section 3, and computed examples that compare these methods are presented in Section 4. Concluding remarks can be found in Section 5.

Throughout this paper $\|\cdot\|$ denotes the Euclidean vector norm or the spectral matrix norm.

2. A restarted generalized Arnoldi-type (RGAT) method for the generalized eigenproblem

We describe a restarted Arnoldi method for the computation of a few, say $1 \leq l \ll n$, of the eigenvalues of largest magnitude and associated eigenvectors of the generalized eigenproblem (1.1). The method also can be applied to compute the l eigenvalues of smallest magnitude and associated eigenvector. The latter application is commented on at end of this section. The method consists of two parts: reducing an available solution subspace, and enlarging an available solution subspace. We will discuss these parts in order.

2.1. Reducing the solution subspace

Let $l \leq p \ll n$ and let $\mathcal{V}_{2p} \subset \mathbb{C}^n$ denote a subspace of dimension $2p$. The columns of the matrix $V_{2p} \in \mathbb{C}^{n\times 2p}$ form an orthonormal basis for this subspace. When we start the computations, the columns of V_{2p} may, for instance, be generated by first determining an $n \times 2p$ matrix with normally distributed random entries with mean zero, and then orthonormalizing the columns of this matrix. We seek to find approximate generalized eigenvectors in the subspace $\mathcal{V}_{2p}$. The Galerkin equations for the approximate eigenvectors are given by

$$V_{2p}^{*}AV_{2p}Y = V_{2p}^{*}BV_{2p}Y\Lambda, \tag{2.1}$$

where the superscript * denotes transposition and complex conjugation, the columns of the matrix $Y_{2p} = [y_1, y_2, \ldots, y_{2p}] \in \mathbb{C}^{2p\times 2p}$ are eigenvectors and the nontrivial entries of

$$\Lambda_{2p} = \mathrm{diag}[\lambda_1, \lambda_2, \ldots, \lambda_{2p}] \in \mathbb{C}^{2p\times 2p} \tag{2.2}$$

are eigenvalues of the matrix pair $\{V_{2p}^*AV_{2p}, V_{2p}^*BV_{2p}\}$. The small size of the problem (2.1) allows the application of the QZ algorithm to the computation of the eigenvalues λ_i and associated unit eigenvectors y_i. We assume the eigenvalues to be ordered so that

$$|\lambda_1| \geq |\lambda_2| \geq \ldots \geq |\lambda_{2p}|.$$

The generalized eigenproblem (1.1) might not have n linearly independent eigenvectors and have infinite eigenvalues. Similarly, the generalized eigenproblem (2.1) might not have $2p$ linearly independent eigenvectors and may have infinite eigenvalues; see, e.g., [18, Chapter 9]. Here we will for simplicity assume that the eigenproblems (1.1) and (2.1) only have bounded eigenvalues, that the $2p$ eigenvectors of (2.1) are linearly independent, and that $|\lambda_p| > |\lambda_{p+1}|$. These restrictions can be removed at the expense of obtaining a more complicated algorithm for computing the desired eigenpairs. Then

$$x_i = V_{2p}y_i, \quad i = 1, 2, \ldots, p, \tag{2.3}$$

are linearly independent Ritz vectors for the generalized eigenproblem (1.1) associated with the p Ritz values λ_i, $i = 1, 2, \ldots, p$, of largest magnitude. We will keep the subspace spanned by these Ritz vectors and discard the subspace spanned by the remaining Ritz vectors. This is the reduction part of the algorithm. It secures that the algorithm never requires storage of more than $2p$ Ritz vectors simultaneously. Orthogonalize and normalize the vectors $y_1, y_2, \ldots, y_p$, and let the vectors so determined form the columns of the matrix $\widetilde{Y} \in \mathbb{C}^{2p \times p}$. Then the columns of the matrix

$$\widetilde{V}_p = V_{2p}\widetilde{Y} \in \mathbb{C}^{n \times p} \tag{2.4}$$

form an orthonormal basis for the retained solution subspace. This concludes the reduction phase of the algorithm.

2.2. Enlarging the solution subspace

Let $x_1, x_2, \ldots, x_p$ denote the Ritz vectors associated with the p Ritz values $\lambda_1, \lambda_2, \ldots, \lambda_p$ of largest magnitude, cf. (2.3). Thus, these λ_i are the first p diagonal entries of the matrix (2.2). Compute the residual vectors

$$r_i = Ax_i - \lambda_i Bx_i, \qquad i = 1, 2, \ldots, p, \tag{2.5}$$

and orthogonalize them, e.g., by the modified Gram-Schmidt process to obtain the orthonormal vectors $q_1, q_2, \ldots, q_p$. The evaluations (2.5) can be carried out simultaneously with the aid of level 3 BLAS to reduce computation time.

We note that the residual vector r_i may overwrite the Ritz vector x_i for $i = 1, 2, \ldots, p$ to reduce the amount of computer storage required. In exact arithmetic, the vectors r_i are orthogonal to the range of the matrix (2.4). To enforce orthogonality in the presence of round-off errors, we explicitly orthogonalize the vectors $q_i = r_i/\|r_i\|$ against range$(\widetilde{V}_p)$, where we assume that all $r_i \neq 0$, and enlarge the retained solution subspace by including these vectors. Thus, let $Q_p = [q_1, q_2, \ldots, q_p] \in \mathbb{C}^{n \times p}$, compute

$$\hat{Q}_p := (I - \widetilde{V}_p\widetilde{V}_p^*)Q_p, \tag{2.6}$$

and normalize each column of $\hat{Q}_p$ to be of unit Euclidean norm. This yields the matrix $\hat{Q}_p = [\hat{q}_1, \hat{q}_2, \dots, \hat{q}_p] \in \mathbb{C}^{n\times p}$. Then define

$$\hat{V}_{2p} = [\widetilde{V}_p, \hat{Q}_p] \in \mathbb{C}^{n\times 2p}. \tag{2.7}$$

The matrix $\hat{V}_{2p}$ in (2.7) allows us to determine a new projection of the generalized eigenproblem (1.1),

$$\hat{V}_{2p}^* A \hat{V}_{2p} \hat{Y} = \hat{V}_{2p}^* B \hat{V}_{2p} \hat{Y} \hat{\Lambda}, \tag{2.8}$$

which is analogous to (2.1). Thus, $\hat{Y} \in \mathbb{C}^{2p\times 2p}$ and

$$\hat{\Lambda} = \mathrm{diag}[\hat{\lambda}_1, \hat{\lambda}_2, \dots, \hat{\lambda}_{2p}] \in \mathbb{C}^{2p\times 2p}.$$

Let the eigenvalues $\hat{\lambda}_1, \hat{\lambda}_2, \dots, \hat{\lambda}_{2p}$ be ordered in decreasing magnitude, and let $\{\hat{y}_i\}_{i=1}^{2p}$ denote the associated unit eigenvectors of (2.8). We keep the p largest eigenvalues of (2.8), which are Ritz values of (1.1), and the associated Ritz vectors $\hat{x}_i = \hat{V}_{2p}\hat{y}_i$, $i = 1, 2, \dots, p$. These Ritz pairs furnish new approximations of the p eigenpairs of (1.1) with eigenvalues of largest magnitude. The computations are terminated when

$$\max_{1\le j\le l} \frac{|\lambda_j - \hat{\lambda}_j|}{|\hat{\lambda}_j|} \le \delta \tag{2.9}$$

for a user-specified tolerance $\delta > 0$, in which case $\{\hat{\lambda}_i, \hat{x}_i\}_{i=1}^{l}$ are accepted as the desired eigenpairs of (1.1). In case (2.9) does not hold, we set $\lambda_i = \hat{\lambda}_i$ and $x_i = \hat{x}_i$ for $i = 1, 2, \dots, p$, retain the solution subspace $\mathrm{span}\{x_1, x_2, \dots, x_p\}$, and carry out another expansion of this space using residual vectors (2.5) as described above. The computations are described by Algorithm 1. Some computations, such as those of step **b**, can be implemented efficiently by using level 3 BLAS.

In the special case when the matrix B in Algorithm 1 is the identity, the algorithm yields approximations of eigenpairs associated with the eigenvalues of largest magnitude of the matrix A. This special case of Algorithm 1 has been applied explicitly or implicitly by Lampe et al. [8, 9, 10].

When approximations of the eigenpairs of (1.1) with the eigenvalues of smallest magnitude are desired, we order the eigenvalues $\{\hat{\lambda}_i\}_{i=1}^{2p}$ of (2.8) according to increasing magnitude. The Ritz pairs $\{\hat{\lambda}_i, \hat{V}_{2p}\hat{y}_i\}_{i=1}^{p}$ of (1.1) then furnish approximations of the p eigenpairs associated with eigenvalues of smallest magnitude. The stopping criterion (2.9) typically has to be modified by removing the denominators $\hat{\lambda}_j$. This method works the best when eigenvalues of smallest magnitude are the smallest eigenvalues. We proceed analogously when the eigenpairs associated with eigenvalues of largest or smallest real parts are desired. Efficient computation of eigenvalues far from the boundary of the convex hull of the spectrum of A may require the use of a preconditioner.

We remark that analogues of Algorithm 1 for the solution of quadratic and palindromic eigenvalue problems easily can be devised. We refer to [1, 18] for discussions of this kind of eigenvalue problems.

Algorithm 1 A restarted generalized Arnoldi-type (RGAT) for the generalized eigenproblems (2.1).

```
Input: Matrices $A, B \in \mathbb{C}^{n\times n}$, $p > 0$, matrix $V_{2p} \in \mathbb{C}^{n\times 2p}$ such that $V_{2p}^{*}V_{2p} = I$, tolerance $\delta > 0$, number of desired eigenpairs $l$.
Initialize: error:= $2\delta$.
    a. Compute the eigenpairs of the generalized eigenproblem $V_{2p}^{*}AV_{2p}Y = V_{2p}^{*}BV_{2p}Y\Lambda$ with $\Lambda = \mathrm{diag}[\lambda_1, \lambda_2, \dots, \lambda_{2p}]$ with the eigenvalues ordered according to $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_{2p}|$ and $Y = [y_1, y_2, \dots, y_{2p}] \in \mathbb{C}^{2p\times 2p}$.
    b. For $j = 1, 2, \dots, p$ do
        $r_j = (A - \lambda_j B)V_{2p}y_j$
    End For
    c. Let the columns of $Q_p \in \mathbb{C}^{n\times p}$ form an orthonormal basis for
      range$\{r_1, r_2, \dots, r_p\}$. (Can be computed by QR factorization.)
    d. Let the columns of $\widetilde{Y} \in \mathbb{C}^{2p\times p}$ form an orthonormal basis for
      range$\{y_1, y_2, \dots, y_p\}$.
    e. $\widetilde{V}_p := V_{2p}\widetilde{Y}$ ( Reducing the solution subspace)
    f. Reorthogonalize $\hat{Q}_p := (I - \widetilde{V}_p\widetilde{V}_p^{*})Q_p$ and normalize the the columns of $\hat{Q}_p$.
    j. Enlarge search space $\hat{V}_{2p} := [\widetilde{V}_p, \hat{Q}_p]$
    k. Compute the eigenpairs of the new generalized eigenproblem $\hat{V}_{2p}^{*}A\hat{V}_{2p}\hat{Y} = \hat{V}_{2p}^{*}B\hat{V}_{2p}\hat{Y}\hat{\Lambda}$ with $\hat{\Lambda} = \mathrm{diag}[\hat{\lambda}_1, \hat{\lambda}_2, \dots, \hat{\lambda}_{2p}]$ and the eigenvalues ordered according to $|\hat{\lambda}_1| \geq |\hat{\lambda}_2| \geq \cdots \geq |\hat{\lambda}_{2p}|$ and $\hat{Y} = [\hat{y}_1, \hat{y}_2, \dots, \hat{y}_{2p}] \in \mathbb{C}^{2p\times 2p}$.
    l. Compute error:= $\max_{1\leq j\leq l} \frac{|\lambda_j - \hat{\lambda}_j|}{|\hat{\lambda}_j|}$. If error$\leq \delta$ then exit, else goto b.
Output: Approximations of the $l$ eigenvalues of largest magnitude and approximations of associated eigenvectors.
.
```

3. Simultaneous reduction of the matrices A and B

This section outlines an approach described by Hoffnung, Li, and Ye [6, 12] for the simultaneous reduction of a pair of matrices $A, B \in \mathbb{C}^{n \times n}$ to a pair of small matrices. This reduction method also has been applied to the solution of linear discrete ill-posed problems by Tikhonov regularization in general form [16]. A generalization that allows the simultaneous reduction of matrix k-tuplets (with $k > 2$) is described in [17].

We review the generalized Arnoldi process for two square matrices $A, B \in \mathbb{C}^{n \times n}$. This process is based on the observation that for an arbitrary unit vector $v_1 \in \mathbb{C}^n$, generically, there is a unitary matrix V such that

$$V^*AV = H_A \equiv [h_{A;i,j}]_{i,j=1}^{n}, \qquad V^*BV = H_B \equiv [h_{B;i,j}]_{i,j=1}^{n} \tag{3.1}$$

with

$$h_{A;i,j} = 0 \text{ for } i \geq 2j+1, \qquad h_{B;i,j} = 0 \text{ for } i \geq 2j+2. \tag{3.2}$$

Thus, H_A and H_B are generalized Hessenberg matrices. The relations (3.1) and (3.2) are the foundation of Algorithm 2; see [6, 12] for details.

The scalar N in Algorithm 2 tracks the number of vectors v_i generated so far during the computations. Let α_k and β_k denote the values of N at the end of steps **2.d** and **2.f**, respectively, when $j = k$. It can be seen that $k \leq \alpha_k \leq \beta_k \leq \alpha_k + 1$. Thus, the first β_k columns of the the matrix V and the leading $\alpha_k \times k$ and $\beta_k \times k$ principal submatrices of the matrices H_A and H_B, respectively, are generated. Generically, Algorithm 2 yields upon completion of k iterations the generalized Arnoldi decompositions

$$\begin{aligned} AV_{(:,1:k)} &= V_{(:,1:\alpha_k)} H_{A\,(1:\alpha_k,1:k)}, \qquad (3.3)\\ BV_{(:,1:k)} &= V_{(:,1:\beta_k)} H_{B\,(1:\beta_k,1:k)}; \qquad (3.4)\end{aligned}$$

see [6, 12]. Here and throughout, we use MATLAB-like notation $X_{(i:j,k:\ell)}$ to denote the submatrix of X, consisting of the intersections of rows i to j and columns k to ℓ; when $i : j$ is replaced by :, it means all rows, similarly for columns. We note that while $H_{A\,(1:\alpha_k,1:k)}$ and $H_{B\,(1:\beta_k,1:k)}$ are lower banded matrices, their lower bandwidths grow linearly with k. The computations are terminated at Line **2.a** in the (rare) event that the recursions for Algorithm 2 break down. In this case, we obtain the first N columns of V with $\alpha_N = \beta_N = N$. This is a benign breakdown as we obtain the decompositions

$$\begin{aligned} AV_{(:,1:N)} &= V_{(:,1:N)} H_{A\,(1:N,1:N)},\\ BV_{(:,1:N)} &= V_{(:,1:N)} H_{B\,(1:N,1:N)},\end{aligned}$$

with the column space of $V_{(:,1:N)}$ being an invariant subspace of both the matrices A and B.

An examination of Algorithm 2 shows that the set $\{v_1, v_2, \ldots, v_\ell\}$ is generated from the sequence of vectors obtained by multiplying v_1 by A and B in a periodic fashion,

$$\begin{array}{ll} \text{Group 0:} & v_1, \\ \text{Group 1:} & Av_1, Bv_1, \\ \text{Group 2:} & A^2v_1, BAv_1, ABv_1, B^2v_1, \\ \text{Group 3:} & A^3v_1, BA^2v_1, ABAv_1, B^2Av_1, A^2Bv_1, BABv_1, AB^2v_1, B^3v_1, \\ \vdots & \vdots \end{array} \tag{3.5}$$

Algorithm 2 Generalized Arnoldi-type (GAT) process for matrix pairs $\{A,B\}$

```
**Input:** Matrices $A,B \in \mathbb{C}^{n\times n}$, $v_1$, with $||v_1|| = 1$. Number of desired eigenpairs $l$, number of steps $k \geq l$.
**1.** $N = 1$;
**2.** For  j = 1,2,...,k do
    **a.** If $j > N$ break
    **b.** $\hat{v} = Av_j$
    **c.** For $i = 1,2,\ldots,N$ do
    $h_{a;ij} = v_i^*\hat{v}; \hat{v} = \hat{v} - v_i h_{a;ij}$
    End For
    $h_{a;N+1,j} = ||\hat{v}||$
    **d.** If $h_{a;N+1,j} > 0$,
    $N = N+1, v_N = \hat{v}/h_{a;Nj}$
    End If
    $\hat{v} = Bv_j$
    **e.** For $i = 1,2,\ldots,N$ do
    $h_{b;ij} = v_i^*\hat{v}; \hat{v} = \hat{v} - v_i h_{b;ij}$
    End For
    $h_{b;N+1,j} = ||\hat{v}||$,
    **f.** If $h_{b;N+1,j>0}$
    $N = N+1, v_N = \hat{v}/h_{b;Nj}$
    End If
**3.** End For
**4.** Compute the $l$ eigenvalues of the largest magnitude and associated eigenvectors $\{\lambda_i, y_i\}$ of the generalized eigenproblem $H_{A\,(1:k,1:k)}y = \lambda H_{B\,(1:k,1:k)}y$, where $\|y_i\| = 1$, $H_{A\,(1:k,1:k)} = [h_{a;ij}]_{i,j=1}^k$ and $H_{B\,(1:k,1:k)} = [h_{b;ij}]_{i,j=1}^k$.
**Output:** Ritz value and Ritz vector pairs $\{\lambda_i, x_i\}_{i=1}^l$, where $x_i = V_{(:,1:k)}y_i$ and $V_{(:,1:k)} = [v_1, v_2, \ldots, v_k]$.
```

.

ordered from top to bottom and from left to right. α_j (and β_j as well) can only increase by 2 at each step.

For the Ritz value/Ritz vector pairs obtained, we have the following residual bounds generalizing similar results for the Arnoldi or the generalized Arnoldi algorithms.

Theorem 1. *Let λ_i and x_i be the Ritz value and Ritz vector x_i obtained by k steps of the Generalized Arnoldi-type process (Algorithm 2). Let p be the smallest integer such that* $\max\{\alpha_p, \beta_p\} \geq k$. *Then, we have*

$$\begin{aligned} \|(A-\lambda_i B)x_i\| &\leq (\|H_{A\,(k+1:\alpha_k,p:k)}\| + |\lambda_i|\|H_{B\,(k+1:\beta_k,p:k)}\|)\|y_{i\,(p:k)}\| \\ &\leq (\|A\| + |\lambda_i|\|B\|)\|y_{i\,(p:k)}\|. \end{aligned}$$

Proof: It follows from (3.4) that

$$\begin{aligned} (A-\lambda_i B)x_i &= (AV_{(:,1:k)} - \lambda_i BV_{(:,1:k)})y_i \\ &= (V_{(:,1:\alpha_k)}H_{A\,(1:\alpha_k,1:k)} - \lambda_i V_{(:,1:\beta_k)}H_{B\,(1:\beta_k,1:k)})y_i \\ &= (V_{(:,1:k)}H_{A\,(1:k,1:k)} - \lambda_i V_{(:,1:k)}H_{B\,(1:k,1:k)})y_i \\ &\quad +(V_{(:,k+1:\alpha_k)}H_{A\,(k+1:\alpha_k,1:k)} - \lambda_i V_{(:,k+1:\beta_k)}H_{B\,(k+1:\beta_k,1:k)})y_i \\ &= (V_{(:,k+1:\alpha_k)}H_{A\,(k+1:\alpha_k,1:k)} - \lambda_i V_{(:,k+1:\beta_k)}H_{B\,(k+1:\beta_k,1:k)})y_i. \end{aligned}$$

Since $h_{a;ij} = 0$ for $i > \alpha_j$ and $\alpha_{p-1} \leq k$, we have $H_{A\,(k+1:\alpha_k,1:p-1)} = 0$. Similarly, $H_{B\,(k+1:\alpha_k,1:p-1)} = 0$. Taking the norm, we obtain

$$\begin{aligned} \|(A-\lambda_i B)x_i\| &\leq \|(V_{(:,k+1:\alpha_k)}H_{A\,(k+1:\alpha_k,1:k)} - \lambda_i V_{(:,k+1:\beta_k)}H_{B\,(k+1:\beta_k,1:k)})y_i\| \\ &\leq \|H_{A\,(k+1:\alpha_k,1:k)}y_i\| + |\lambda_i|\|H_{B\,(k+1:\beta_k,1:k)}y_i\| \\ &= \|H_{A\,(k+1:\alpha_k,p:k)}y_{i\,(p:k)}\| + |\lambda_i|\|H_{B\,(k+1:\beta_k,p:k)}y_{i\,(p:k)}\| \\ &\leq (\|H_{A\,(k+1:\alpha_k,p:k)}\| + |\lambda_i|\|H_{B\,(k+1:\beta_k,p:k)}\|)\|y_{i\,(p:k)}\| \\ &\leq (\|A\| + |\lambda_i|\|B\|)\|y_{i\,(p:k)}\|, \end{aligned}$$

where we have used that $\|y_i\| = 1$ and

$$\|H_{A\,(k+1:\alpha_k,p:k)}\| \leq \|H_{A\,(1:\alpha_k,1:k)}\| = \|V^*_{(:,1:\alpha_k)}AV_{(:,1:k)}\| \leq \|A\|,$$

and similarly $\|H_{B\,(k+1:\beta_k,p:k)}\| \leq \|B\|$. □

The bound shows that we may expect a small residual in two ways. If $\|H_{A\,(k+1:\alpha_k,p:k)}\|$ and $\|H_{B\,(k+1:\beta_k,p:k)}\|$ are small, then the residual is small. In this case, we have a nearly invariant subspace and all Ritz values are accurate approximations of eigenvalues. On the other hand, when the last $k-p+1$ components of y_i are small, the residual also is small. In this case, the Ritz values provide accurate approximations of eigenvalues, without obtaining an invariant subspace.

4. Numerical examples

This section presents a few computed examples that illustrate the performance of Algorithms 1 and 2. The former implements the RGAT (restarted generalized Arnoldi-type) method and the latter the GAT (generalized Arnoldi-type) method. We allow both algorithms about the same storage space during the computations. This is meaningful when the matrices A and B are large and the available computer storage is limited. The RGAT method requires storage space for the matrices V_{2p} and AV_{2p}; hence the total storage requirements is about $4p$ n-vectors in addition to storage required for functions for the evaluation of matrix-vector products with the matrices A and B. The GAT method requires storage of about $2k$ n-vectors in addition to storage required the evaluation of matrix-vector products with A and B. Here k is the number of iterations carried out with Algorithm 2. We let $k = 2p$. Then Algorithms 1 and 2 demand about the same storage space. The initial vector for both algorithms is chosen to have normally distributed entries with mean zero and variance one. The matrices used in our experiments are from Matrix Market [13].

All numerical calculations were carried out in MATLAB R2019a on a MacBook Pro computer with a 2.8 GHz Quad-Core Intel Core i7 processor and 16 GB of RAM. The “exact” eigenvalues and eigenvectors in our experiments are computed with the MATLAB function eig and sorted appropriately.

Example 1

Let $A, B \in \mathbb{R}^{512\times 512}$ be the nonsymmetric matrices denoted by DWA512 and DWB512 in [13] and referred to as “Square Dielectric Waveguide”. The MATLAB function eig yields accurate approximations of the 5 eigenvalues of largest magnitude:

$$\lambda_1 = -8.2510,\;\; \lambda_2 = -7.4413,\;\; \lambda_3 = -6.7296,\;\; \lambda_4 = -6.5730,\;\; \lambda_5 = -5.8698.$$

We consider these eigenvalues exact.

Set $\delta = 10^{-6}$ and $p = 5$ in the RGAT method to compute $l = 5$ eigenpairs with eigenvalues of largest magnitude. The computed eigenvalues are shown in Table 1. The eigenvalues determined by the RGAT method have tiny imaginary parts, which can be ignored.

Table 1: Example 1: Approximate eigenvalues of largest magnitude computed by the RGAT and GAT methods. i denotes the imaginary unit.

	Eigenvalues	
	λ_{RGAT}	λ_{GAT}
1	$-8.2510 + 1.0802 \cdot 10^{-18} i$	-7.0519
2	$-7.4413 + 3.6125 \cdot 10^{-15} i$	-3.9702
3	$-6.7296 - 1.9641 \cdot 10^{-13} i$	-3.2267
4	$-6.5730 - 2.5972 \cdot 10^{-14} i$	-3.1255
5	$-5.8695 - 1.9169 \cdot 10^{-10} i$	-2.7573

Figure 1 depicts the exact eigenvalues and the computed approximate eigenvalues determined by the RGAT and GAT methods. We can notice that the eigenvalues computed by the RGAT method are very accurate, while the eigenvalues computed by GAT are not.

Table 2 displays the relative error in the eigenvalue approximations computed by the RGAT and GAT methods. We denote these errors by E_{RGAT} and E_{GAT}, respectively.

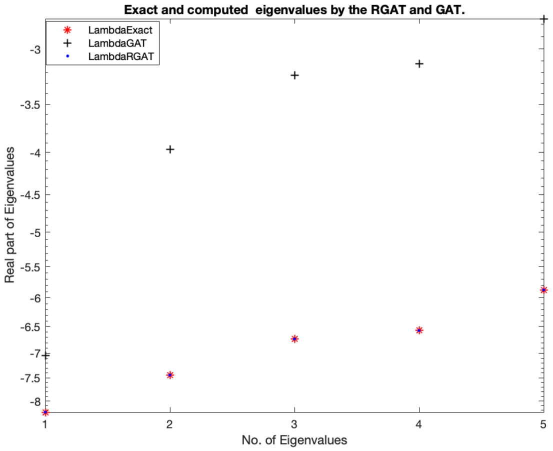

Figure 1: Example 1.

The table shows the RGAT method to yield more accurate approximations than the GAT method. The accuracy of the eigenvalues of largest magnitude is the highest. We will return to this observation below.

Table 2: Example 1: Relative error in the computed approximate eigenvalues determined by the RGAT and GAT methods.

E_{RGAT}	E_{GAT}
$1.4424 \cdot 10^{-14}$	$1.4532 \cdot 10^{-1}$
$2.6665 \cdot 10^{-10}$	$4.6647 \cdot 10^{-1}$
$5.2484 \cdot 10^{-9}$	$5.2052 \cdot 10^{-1}$
$2.9322 \cdot 10^{-9}$	$5.2449 \cdot 10^{-1}$
$5.7368 \cdot 10^{-5}$	$5.3025 \cdot 10^{-1}$

Table 3 shows the relative errors in the approximate eigenvectors computed by the RGAT and GAT methods. We denote these errors by EV_{RGAT} and EV_{GAT}.

While the storage requirement of the RGAT and GAT methods is about the same, the computational effort is not. When $\delta = 10^{-6}$, the RGAT method requires 2280 matrix-vector product evaluations, and the number of iteration is 114. If instead $\delta = 10^{-4}$, then the numbers of matrix-vector product evaluations required by RGAT is 1240 and the number of iteration is 62. The number of matrix-vector product evaluations of the GAT method is only 20. We conclude that the RGAT method is competitive when the matrices are large and computer storage is a limited.

Table 4 displays how the number of iterations changes with p. Just like above, we set $\delta = 10^{-6}$ and are interested in determining the $l = 5$ eigenvalues of largest magnitude. The table shows the number of iterations required to satisfy the stopping

Table 3: Example 1: Relative error in the approximate eigenvectors computed by the RGAT and GAT methods.

EV_{RGAT}	EV_{GAT}
$8.1953 \cdot 10^{-11}$	$8.7019 \cdot 10^{-1}$
$3.0936 \cdot 10^{-8}$	$1.3226 \cdot 10^{0}$
$1.0689 \cdot 10^{-5}$	$1.4173 \cdot 10^{0}$
$1.8211 \cdot 10^{-5}$	$1.4132 \cdot 10^{0}$
$2.7825 \cdot 10^{-2}$	$1.3457 \cdot 10^{0}$

criterion of Algorithm 1 to decrease as p increases. Thus, when l eigenvalues of largest magnitude are desired, and there is enough computer storage available to allow $p > l$, the number of iterations typically will decrease by increasing p. The computational effort per iteration grows with p. The choice of p that results in the shortest execution time depends on properties of the matrix pair $\{A, B\}$, including the size of the matrices, the number of desired eigenvalues l, implementation details of the Algorithm 1, as well as on the properties of the computer used.

Table 4: Example 1: Number of iterations as a function of with p when we seek to determine the $l = 5$ largest eigenvalues.

p	Number of iterations
5	114
7	56
10	34

Example 2

Let $A, B \in \mathbb{R}^{782 \times 782}$ be the nonsymmetric matrices BFW782A and BFW782B from [13]. They are referred to as "Bounded Finline Dielectric Waveguide". The MATLAB function eig determines the five eigenvalues of largest magnitude

$$\lambda_1 = -2.7557 \cdot 10^6,\ \ \lambda_2 = -2.7383 \cdot 10^6,\ \ \lambda_3 = -2.6701 \cdot 10^6,$$
$$\lambda_4 = -2.5707 \cdot 10^6,\ \ \lambda_5 = -2.4464 \cdot 10^6.$$

We consider these eigenvalues exact. We are interested in computing approximations of these eigenvalues and associated eigenvectors by the RGAT and GAT methods; see Table 5 for the eigenvalues. The RGAT method is applied with $\delta = 10^{-6}$.

Figure 2 displays the exact and computed eigenvalues by the RGAT and GAT methods. The eigenvalues computed by the RGAT method are seen to be very close to the exact eigenvalues.

Table 6 displays the relative errors in the eigenvalues computed by the RGAT and GAT methods, which we designate by E_{RGAT} and E_{GAT}, respectively, and Table 7 shows the relative error in eigenvectors computed by the RGAT and GAT methods.

Table 5: Example 2: Approximations of eigenvalues of largest magnitude computed by the RGAT and GAT methods. *i* denotes the imaginary unit.

Eigenvalues		
	λ_{RGAT}	λ_{GAT}
1	$-2.7557 \cdot 10^{6} - 1.5176 \cdot 10^{-10} i$	$-2.8261 \cdot 10^{4}$
2	$-2.7383 \cdot 10^{6} - 5.3305 \cdot 10^{-11} i$	$-3.7397 \cdot 10^{4}$
3	$-2.6701 \cdot 10^{6} - 4.2551 \cdot 10^{-10} i$	$-6.8041 \cdot 10^{4}$
4	$-2.5707 \cdot 10^{6} + 1.8653 \cdot 10^{-10} i$	$-8.9053 \cdot 10^{4}$
5	$-2.4468 \cdot 10^{6} + 9.9294 \cdot 10^{-10} i$	$-1.1114 \cdot 10^{5}$

Table 6: Example 2: Relative error in the approximate eigenvalues computed by the RGAT and GAT methods.

E_{RGAT}	E_{GAT}
$2.3709 \cdot 10^{-6}$	$9.8974 \cdot 10^{-1}$
$1.4843 \cdot 10^{-6}$	$9.8634 \cdot 10^{-1}$
$3.8773 \cdot 10^{-6}$	$9.7452 \cdot 10^{-1}$
$2.7669 \cdot 10^{-6}$	$9.6536 \cdot 10^{-1}$
$1.3801 \cdot 10^{-4}$	$9.5457 \cdot 10^{-1}$

Table 7: Example 2: Relative error in the approximate eigenvectors computed by the RGAT and GAT methods.

EV_{RGAT}	EV_{GAT}
$2.7784 \cdot 10^{-4}$	$1.4087 \cdot 10^{0}$
$1.2557 \cdot 10^{-4}$	$1.4208 \cdot 10^{0}$
$3.6269 \cdot 10^{-4}$	$1.3647 \cdot 10^{0}$
$7.8851 \cdot 10^{-3}$	$1.3953 \cdot 10^{0}$
$3.8769 \cdot 10^{-2}$	$1.4079 \cdot 10^{0}$

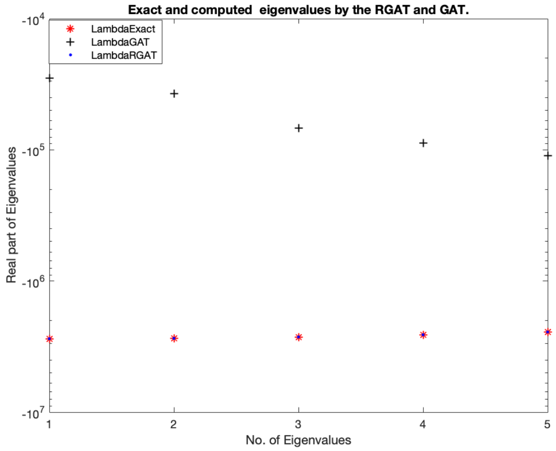

Figure 2: Example 2.

The number of matrix-vector product evaluations required by RGAT is 3860 when $\delta = 10^{-6}$ and the number of iterations is 193. Increasing δ reduces the number of iterations. For instance, when $\delta = 10^{-4}$, the number of matrix-vector product evaluations by the RGAT method is reduced to 2060 and the number of iterations to 103.

Table 8 shows the number of iterations of the RGAT method decreases as p increases, and the parameter $\delta = 10^{-6}$ and the number of desired eigenvalues $l = 5$ are kept fixed.

Table 8: Example 2: The number of iterations as a function of p.

p	Number of iterations
5	193
7	153
10	121

5. Conclusion

We have developed a generalized Arnoldi method for the calculation of a few eigenpairs of a generalized eigenvalue problem. The method requires a fairly small amount of computer memory, and does not require the solution of large linear systems of equations. However, many iterations may be needed. The number of iterations depends on the required accuracy, on the amount of storage used, and of course on the matrix pair $\{A,B\}$.

Acknowledgment

The authors would like to thank a referee for insightful comments. M.A. would like to thank the Deanship of Scientific Research at Qassim University for funding his research for this project. Research of L.R. was supported in part by NSF grant DMS-1720259, and research of Q.Y. was supported in part by NSF grant DMS-1821144.

References

[1] S. S. Ahmad, *Perturbation analysis for palindromic and anti-palindromic nonlinear eigenvalue problems*, Electron. Trans. Numer. Anal., 51 (2019), pp. 151–168.

[2] J. Baglama, D. Calvetti, and L. Reichel, *IRBL: An implicitly restarted block Lanczos method for large-scale Hermitian eigenproblems*, SIAM J. Sci. Comput., 24 (2003), pp. 1650–1677.

[3] T. Ericsson and A. Ruhe, *The spectral transformation Lanczos method for the numerical solution of large sparse generalized symmetric eigenvalue problems*, Math. Comp., 35 (1980), pp. 1251–1268.

[4] G. Golub and Q. Ye, *An inverse free preconditioned Krylov subspace method for symmetric generalized eigenvalue problems*, SIAM J. Sci. Comput., 24 (2002), pp. 312–334.

[5] R. G. Grimes, J. G. Lewis, and H. D. Simon, *A shifted block Lanczos algorithm for solving sparse symmetric generalized eigenproblems*, SIAM J. Matrix Anal., 15 (1994), pp. 228–272.

[6] L. Hoffnung, R.-C. Li, and Q. Ye, *Krylov type subspace methods for matrix polynomials*, Linear Algebra Appl., 415 (2006), pp. 52–81.

[7] Z. Jia and Y. Zhang, *A refined shift-and-invert Arnoldi algorithm for large unsymmetric generalized eigenproblems*, Comput. Math. Appl., 44 (2002), pp. 1117–1127.

[8] J. Lampe, M. Rojas, D. C. Sorensen, and H. Voss, *Accelerating the LSTRS algorithm*, SIAM J. Sci. Comput., 33 (2011), pp. 175–194.

[9] J. Lampe and H. Voss, *A fast algorithm for solving regularized total least squares problems*, Electron. Trans. Numer. Anal., 31 (2008), pp. 12–24.

[10] J. Lampe and H. Voss, *Efficient determination of the hyperparameter in regularized total least squares*, Appl. Numer. Math., 62 (2012), pp. 1229–1241.

[11] J. Lampe and H. Voss, *A survey on variational characterizations for nonlinear eigenvalue problems*, Electron. Trans. Numer. Anal., 55 (2022), pp. 1–75.

[12] R.-C. Li and Q. Ye, *A Krylov subspace method for quadratic matrix polynomials with application to constrained least squares problems*, SIAM J. Matrix Anal. Appl., 25 (2003), pp. 405–428.

[13] Matrix Market, *A repository of test data for use in comparative studies of algorithms for numerical linear algebra*. Available at URL http://math.nist.gov/MatrixMarket/.

[14] M. M. Pandur, *Preconditioned gradient iterations for the eigenproblem of definite matrix pairs*, Electron. Trans. Numer. Anal., 51 (2019), pp. 331–362.

[15] P. Quillen and Q. Ye, *A block inverse-free preconditioned Krylov subspace method for symmetric generalized eigenvalue problems*, J. Comput. Appl. Math., 233 (2010), pp. 1298–1313.

[16] L. Reichel, F. Sgallari, and Q. Ye, *Tikhonov regularization based on generalized Krylov subspace methods*, Appl. Numer. Math., 62 (2012), pp. 1215–1228.

[17] L. Reichel and X. Yu, *Tikhonov regularization via flexible Arnoldi reduction*, BIT Numer. Math., 55 (2015), pp. 1145–1168.

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

[19] G. L. G. Sleijpen, A. G. L. Booten, D. R. Fokkema, and H. A. Van der Vorst, *Jacobi-Davidson type methods for generalized eigenproblems and polynomial eigenproblems*, BIT Numer. Math., 36 (1996), pp. 595–633.

[20] D. C. Sorensen, *Numerical methods for large eigenvalue problems*, Acta Numer., 11 (2002), pp. 519–584.

[21] H. Voss, *An Arnoldi method for nonlinear eigenvalue problems*, BIT Numer. Math., 44 (2004), pp. 387–401.