mathematics

Article

Weighted Block Golub-Kahan-Lanczos Algorithms for Linear Response Eigenvalue Problem

Hongxiu Zhong [1,*], Zhongming Teng [2] and Guoliang Chen [3]

[1] School of Science, Jiangnan University, Wuxi 214122, China
[2] College of Computer and Information Science, Fujian Agriculture and Forestry University, Fuzhou 350002, China; peter979@163.com
[3] School of Mathematical Sciences, Shanghai Key Laboratory of PMMP, East China Normal University, Shanghai 200241, China; glchen@math.ecnu.edu.cn
[*] Correspondence: zhonghongxiu@126.com

Received: 30 November 2018; Accepted: 29 December 2018; Published: 7 January 2019

Abstract: In order to solve all or some eigenvalues lied in a cluster, we propose a weighted block Golub-Kahan-Lanczos algorithm for the linear response eigenvalue problem. Error bounds of the approximations to an eigenvalue cluster, as well as their corresponding eigenspace, are established and show the advantages. A practical thick-restart strategy is applied to the block algorithm to eliminate the increasing computational and memory costs, and the numerical instability. Numerical examples illustrate the effectiveness of our new algorithms.

Keywords: linear response eigenvalue problem; block methods; weighted Golub-Kahan-Lanczos algorithm; convergence analysis; thick restart

AMS Subject Classification: 65F15; 15A18

1. Introduction

In this paper, we are interested in solving the linear response eigenvalue problem (LREP):

$$\mathbf{H}z := \begin{bmatrix} 0 & M \\ K & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \lambda \begin{bmatrix} u \\ v \end{bmatrix} = \lambda z,$$

where K and M are $N \times N$ real symmetric positive definite matrices. Such a problem arises from studying the excitation energy of many particle systems in computational quantum chemistry and physics [1–3]. It also known as the Bethe-Salpeter (BS) eigenvalue-problem [4] or the random phase approximation (RPA) eigenvalue problem [5]. There has immense past and recent work in developing efficient numerical algorithms and attractive theories for LREP [6–15].

Since all the eigenvalues of $\mathbf{H}$ are real nonzero and appear in pairs $\{\lambda, -\lambda\}$ [6], thus we order the eigenvalues in ascending order, i.e.,

$$-\lambda_1 \leq \cdots \leq -\lambda_N < \lambda_N \leq \cdots \leq \lambda_1.$$

In this paper, we focus on a small portion of the positive eigenvalues for LREP, i.e., λ_i, $i = k, k+1, \cdots, \ell$ with $1 \leq k \leq \ell \leq N$ and $\ell - k + 1 \ll N$, and their corresponding eigenvectors. We only consider the real case, all the results can be easily applied to the complex case.

The weighted Golub-Kahan-Lanczos method (**wGKL**) for LREP was introduced in [16]. It produces recursively a much small projection $\mathbf{B}_j = \begin{bmatrix} 0 & B_j \\ B_j^T & 0 \end{bmatrix}$ of $\mathbf{H}$ at j-th iteration, where $B_j \in \mathbb{R}^{j \times j}$

is upper bidiagonal. Afterwards, the eigenpairs of **H** can be constructed by the singular value decomposition of B_j. The convergence analysis performs that running k iterations of **wGKL** is equivalently running $2k$ iterations of a weighted Lanczos algorithm for **H** [16]. Actually, B_j can be also a lower bidiagonal matrix, and the same discussion can be taken place as in the case of B_j is upper bidiagonal. In the following, we only consider the upper bidiagonal case.

It is well known that the single-vector Lanczos method is widely used for searching a small number of extreme eigenvalues, and it may encounter very slow convergence when the wanted eigenvalues stay in a cluster [17]. Instead, a block Lanczos method with a suitable block size is capable of computing a cluster of eigenvalues including multiple eigenvalues very quickly. Motivated by this idea, we are going to develop a block version of **wGKL** in [16] in order to find efficiently all or some positive eigenvalues within a cluster for LREP. Based on the standard block Lanczos convergence theory in [17], the error bounds of approximation to an eigenvalue cluster, as well as their corresponding eigenspace are established to illustrate the advantage of our weighted block Golub-Kahan-Lanczos algorithm (**wbGKL**).

As the increasing size of the Krylov subspace, the storage demands, computational costs, and numerical stability of a simple version of a block Lanczos method may be affected [18]. Several kinds of efficiently restarting strategies to eliminate these effects are developed for the classic Lanczos method, such as, implicitly restart method [19], thick restart method [20]. In order to make our block method more practical, and using the special structure of LREP, we consider the thick restart strategy to our block method.

The rest of this paper is organized as follows. Section 2 gives some necessary preliminaries for our later use. In Section 3, the weighted block Golub-Kahan-Lanczos algorithm (**wbGKL**) for LREP is presented, and its convergence analysis is discussed. Section 4 proposed the thick restart weighted block Golub-Kahan-Lanczos algorithm (**wbGKL-TR**). The numerical examples are tested in Section 5 to illustrate the efficiency of our new algorithms. Finally, some conclusions are given in Section 6.

Throughout this paper, $\mathbb{R}^{m\times n}$ is the set of all $m \times n$ real matrices, $\mathbb{R}^n = \mathbb{R}^{n\times 1}$, and $\mathbb{R} = \mathbb{R}^1$. I_n (or simply I if its dimension is clear from the context) is the $n \times n$ identity matrix, and $0_{m\times n}$ is an $m \times n$ matrix of zero. The superscript "^{T}" denotes transpose. $\|\cdot\|_F$ denotes the Frobenius norm of a matrix, and $\|\cdot\|_2$ denotes the 2-norm of a matrix or a vector. For a matrix $X \in \mathbb{R}^{m\times n}$, $rank(X)$ denotes the rank of X, and $\mathcal{R}(X) = span(X)$ denotes the column space of X; the submatrices $X_{i:j,:}$ and $X_{:,k:\ell}$ of X composed by the intersections of row i to row j and column k to column ℓ, respectively. For matrices or scalars X_i, $diag(X_1, \cdots, X_k)$ denotes the block diagonal matrix with the i-th diagonal block X_i.

2. Preliminaries

For a symmetric positive definite matrix $W \in \mathbb{R}^{N\times N}$, the W-inner product is defined as following

$$\langle x, y\rangle_W := y^T W x, \quad \forall x, y \in \mathbb{R}^N.$$

If $\langle x, y\rangle_W = 0$, then we denote it by $x \perp_W y$, and call it with x and y are W-orthogonal. The projector Π_W is called the W-orthogonal projector onto $\mathcal{Y}$ if for any $y \in \mathbb{R}^N$,

$$\Pi_W y \in \mathcal{Y}, \quad (I - \Pi_W)y \perp_W \mathcal{Y}.$$

For two subspaces $\mathcal{X}, \mathcal{Y} \subseteq \mathbb{R}^N$, and suppose $k = dim(\mathcal{X}) \leq dim(\mathcal{Y}) = \ell$, if $X \in \mathbb{R}^{N\times k}$ and $Y \in \mathbb{R}^{N\times \ell}$ are W-orthonormal basis of $\mathcal{X}$ and $\mathcal{Y}$, respectively, i.e.,

$$X^T W X = I_k,\ \mathcal{X} = \mathcal{R}(X) \quad \text{and} \quad Y^T W Y = I_\ell,\ \mathcal{Y} = \mathcal{R}(Y),$$

and ν_j for $j = 1, \cdots, k$ with $\nu_1 \leq \cdots \leq \nu_k$ are the singular values of $Y^T W X$, then the W-canonical angles $\theta_W^{(j)}(\mathcal{X}, \mathcal{Y})$ from $\mathcal{X}$ to $\mathcal{Y}$ are defined by

$$0 \leq \theta_W^{(j)}(\mathcal{X}, \mathcal{Y}) = \arccos \nu_j \leq \pi/2, \quad \text{for } j = 1, \cdots, k.$$

If $k = \ell$, these angles can be said between $\mathcal{X}$ and $\mathcal{Y}$. Obviously, $\theta_W^{(1)}(\mathcal{X}, \mathcal{Y}) \geq \cdots \geq \theta_W^{(k)}(\mathcal{X}, \mathcal{Y})$. Set

$$\Theta_W(\mathcal{X}, \mathcal{Y}) = diag(\theta_W^{(1)}(\mathcal{X}, \mathcal{Y}), \cdots, \theta_W^{(k)}(\mathcal{X}, \mathcal{Y})).$$

Especially, if $k = 1$, X is a vector, there is only one W-canonical angle from $\mathcal{X}$ to $\mathcal{Y}$. In the following, we may use a matrix in one or both arguments of $\Theta_W(\cdot, \cdot)$, i.e., $\Theta_W(X, Y)$ with the understanding that it means the subspace spanned by the columns of the matrix argument.

The following two lemmas are important to our later analysis, and for proofs and more details, the reader is referred to [12,16].

Lemma 1. ([12] Lemma 3.2). *Let $\mathcal{X}$ and $\mathcal{Y}$ be two subspaces in $\mathbb{R}^N$ with equal dimensional $dim(\mathcal{X}) = dim(\mathcal{Y}) = k$. Suppose $\theta_W^{(1)}(\mathcal{X}, \mathcal{Y}) < \pi/2$. Then, for any set $y_1, y_2, \cdots, y_{k_1}$ of the basis vectors in $\mathcal{Y}$ where $1 \leq k_1 \leq k$, there is a set $x_1, x_2, \cdots, x_{k_1}$ of linearly independent vectors in $\mathcal{X}$ such that $\Pi_W x_j = y_j$ for $1 \leq j \leq k_1$, where Π_W is the W-orthogonal projector onto $\mathcal{Y}$.*

Lemma 2. ([16] Proposition 3.1). *The matrix MK has N position eigenvalues $\lambda_1^2 \geq \lambda_2^2 \geq \cdots \geq \lambda_N^2$ with $\lambda_j > 0$. The corresponding right eigenvectors $\xi_1, \cdots, \xi_N$ can be chosen K-orthonormal, and the corresponding left eigenvectors $\eta_1, \cdots, \eta_N$ can be chosen M-orthonormal. In particular, for given $\{\xi_j\}$, one can choose $\eta_j = \lambda_j^{-1} K \xi_j$, and for given $\{\eta_j\}$, $\xi_j = \lambda_j^{-1} M \eta_j$, for $j = 1, \cdots, N$.*

3. Weighted Block Golub-Kahan-Lanczos Algorithm

3.1. Weighted Block Golub-Kahan-Lanczos Algorithm

In this section, we plan to introduce the weighted block Golub-Kahan-Lanczos algorithm (**wbGKL**) for LREP, which is a block version of the weighted Golub-Kahan-Lanczos algorithm [16]. Algorithm 1 gives the process of recursively generating the M-orthonormal matrix $\mathcal{X}_n$, the K-orthonormal matrix $\mathcal{Y}_n$, and the block bidiagonal matrix $\mathcal{B}_n$. Giving $Y_1 \in \mathbb{R}^{n \times n_b}$ with $Y_1^T K Y_1 = I_{n_b}$, denoting $E_n^T = [0_{n_b \times (n-1)n_b}, I_{n_b}]$, and

$$\mathcal{X}_n = [X_1, \cdots, X_n], \;\; \mathcal{Y}_n = [Y_1, \cdots, Y_n], \;\; \mathcal{B}_n = \begin{bmatrix} A_1 & B_1 & & \\ & A_2 & \ddots & \\ & & \ddots & B_{n-1} \\ & & & A_n \end{bmatrix},$$

then we have the relation from Algorithm 1:

$$K\mathcal{Y}_n = \mathcal{X}_n \mathcal{B}_n, \quad M\mathcal{X}_n = \mathcal{Y}_n \mathcal{B}_n^T + Y_{n+1} B_n^T E_n^T, \tag{1}$$

and

$$\mathcal{X}_n^T M \mathcal{X}_n = I_{nn_b} = \mathcal{Y}_n^T K \mathcal{Y}_n.$$

Remark 1. *In Algorithm 1, we only consider the case that $rank(\widetilde{X}_j) = rank(\widetilde{Y}_{j+1}) = n_b$, no further treatment is provided for the cases $rank(\widetilde{X}_j) < n_b$ or $rank(\widetilde{Y}_{j+1}) < n_b$. Because K and M are both symmetric positive definite, thus the two W in* **Step 2** *are both reversible.*

Algorithm 1: wbGKL

```
1. Choose $Y_1$ satisfying $Y_1^T K Y_1 = I_{n_b}$, and set $W = I_{n_b}$, $B_0 = I_{n_b}$, $X_0 = 0_{n\times n_b}$. Compute
$F = KY_1$.
2. For $j = 1, 2, \cdots, n$
    $\widetilde{X}_j = FW - X_{j-1}B_{j-1}$
    $F = M\widetilde{X}_j$
    Do Cholesky decomposition $\widetilde{X}_j^T F = W^T W$
    $A_j = W, W = inv(W), X_j = \widetilde{X}_j W$        $\%W = inv(W)$ means $W = W^{-1}$
    $\widetilde{Y}_{j+1} = FW - Y_j A_j^T$
    $F = K\widetilde{Y}_{j+1}$
    Do Cholesky decomposition $\widetilde{Y}_{j+1}^T F = W^T W$
    $B_j = W^T, W = inv(W), Y_{j+1} = \widetilde{Y}_{j+1} W$
  End
```

Remark 2. *With j increasing in* **Step 2**, *the M-orthogonality of $\mathcal{X}_j$ and the K-orthogonality of $\mathcal{Y}_j$ will slowly lose. Thus, in practice, we can add a re-orthogonalization process in each iteration to eliminate the defect. The same strategy is executed in the following algorithms.*

From (1), we have

$$\begin{bmatrix} 0 & M \\ K & 0 \end{bmatrix} \begin{bmatrix} \mathcal{Y}_n & 0 \\ 0 & \mathcal{X}_n \end{bmatrix} = \begin{bmatrix} \mathcal{Y}_n & 0 \\ 0 & \mathcal{X}_n \end{bmatrix} \begin{bmatrix} 0 & \mathcal{B}_n^T \\ \mathcal{B}_n & 0 \end{bmatrix} + \begin{bmatrix} Y_{n+1} \\ 0 \end{bmatrix} B_n^T E_{2n}^T$$

with $E_{2n}^T = [0_{n_b\times(2n-1)n_b}, I_{n_b}]$. Then, the approximate eigenpairs of $\mathbf{H}$ can be obtained by solving a small eigenvalue problem of $\left[\begin{smallmatrix} 0 & \mathcal{B}_n^T \\ \mathcal{B}_n & 0 \end{smallmatrix}\right]$. Suppose $\mathcal{B}_n$ has an singular value decomposition

$$\mathcal{B}_n = \Phi \Sigma_n \Psi^T, \tag{2}$$

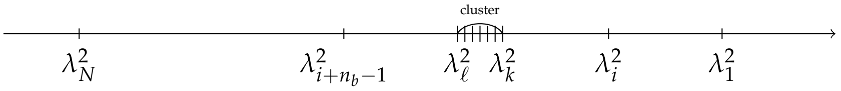
where $\Phi = [\phi_1, \phi_2, \cdots, \phi_{nn_b}]$, $\Psi = [\psi_1, \psi_2, \cdots, \psi_{nn_b}]$, $\Sigma_n = [\sigma_1, \sigma_2, \cdots, \sigma_{nn_b}]$ with $\sigma_1 \geq \cdots \geq \sigma_{nn_b} > 0$. Thus, we can take $\pm\sigma_j (1 \leq j \leq nn_b)$ as the Ritz values of $\mathbf{H}$ and

$$\tilde{z}_j = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathcal{Y}_n \psi_j \\ \pm\mathcal{X}_n \phi_j \end{bmatrix}, \quad 1 \leq j \leq nn_b,$$

as the corresponding $\mathbf{K}$-orthonormal Ritz vectors, where $\mathbf{K} = \left[\begin{smallmatrix} K & 0 \\ 0 & M \end{smallmatrix}\right]$.

3.2. Convergence Analysis

In this section, we first consider the convergence analysis when using the first few σ_j as approximations to the first few λ_j. Then, the similar theories are presented if using the last few σ_j as approximations to the last few λ_j. Since a block Lanczos method with a suitable block size which is not smaller than the size of an eigenvalue cluster can compute all eigenvalues in the cluster. Now, we are considering the i-th to $(i + n_b - 1)$-st eigenpairs of LREP, in which the k-th to ℓ-th eigenvalues form a cluster as in the following figure with $1 \leq i \leq k \leq \ell \leq i + n_b - 1 \leq nn_b$ and $k \leq n$.



Here, the squares of the eigenvalues for LREP are listed. Hence, motivated by [12,17], we analyze the convergence of the cluster eigenvalues and their corresponding eigenspace, and give the error bounds of the approximate eigenpairs belonging to eigenvalue cluster together, instead of separately for each individual eigenpair.

We first give some notations and equations, which are critical in our main theorem. Note that from (1), we get

$$MK\mathcal{Y}_n = \mathcal{Y}_n\mathcal{B}_n^T\mathcal{B}_n + Y_{n+1}B_n^T A_n E_n^T. \tag{3}$$

Since (2) is the singular value decomposition of $\mathcal{B}_n$, thus the eigenvalues of $\mathcal{B}_n^T\mathcal{B}_n$ are σ_j^2 with the associated eigenvectors ψ_j for $1 \le j \le nn_b$.

From Lemma 2, if we let $\Xi = [\xi_1, \cdots, \xi_N]$, and $\Gamma = [\eta_1, \cdots, \eta_N]$, then $\Gamma = K\Xi\Lambda^{-1}$, and

$$MK\Xi = \Xi\Lambda^2. \tag{4}$$

Write Ξ and Λ^2 as

$$\Xi = \begin{array}{c} \begin{array}{ccc} {\scriptstyle i-1} & {\scriptstyle n_b} & {\scriptstyle N-n_b-i+1} \end{array} \\ \left[\begin{array}{ccc} \Xi_1 & \Xi_2 & \Xi_3 \end{array}\right] \end{array}, \qquad \Lambda^2 = \begin{array}{cc} & \begin{array}{ccc} {\scriptstyle i-1} & {\scriptstyle n_b} & {\scriptstyle N-n_b-i+1} \end{array} \\ \begin{array}{c} {\scriptstyle i-1} \\ {\scriptstyle n_b} \\ {\scriptstyle N-n_b-i+1} \end{array} & \begin{bmatrix} \Lambda_1^2 & & \\ & \Lambda_2^2 & \\ & & \Lambda_3^2 \end{bmatrix} \end{array}.$$

Let $\check{\Xi}_2 = \Xi_{(:,k:\ell)}$ and $\check{\Lambda}_2^2 = diag(\lambda_k^2, \cdots, \lambda_\ell^2)$. Denote C_j the first kind Chebyshev polynomial with j-th degree, and $0 \le j \le n$.

In the following, we assume

$$\theta_K^{(1)}(Y_1, \Xi_2) < \pi/2, \tag{5}$$

i.e., $rank(Y_1^T K\Xi_2) = n_b$, then from Lemma 1, we have $\exists\ Z \in \mathbb{R}^{N\times(\ell-k+1)}$ with $\mathcal{R}(Z) \subseteq \mathcal{R}(Y_1)$, s.t.,

$$\Xi_2\Xi_2^T KZ = \check{\Xi}_2. \tag{6}$$

Theorem 1. *Suppose $\theta_K^{(1)}(Y_1, \Xi_2) < \pi/2$, and Z satisfy (6), then we have*

$$\|diag(\lambda_k^2 - \sigma_k^2, \cdots, \lambda_\ell^2 - \sigma_\ell^2)\|_F \le (\lambda_k^2 - \lambda_N^2)\frac{\pi_{i,k,\ell}^2}{C_{n-k}^2(1+2\gamma_{i,\ell})}\|\tan^2\Theta_K(\check{\Xi}_2, Z)\|_F \tag{7}$$

with

$$\gamma_{i,\ell} = \frac{\lambda_\ell^2 - \lambda_{i+n_b}^2}{\lambda_{i+n_b}^2 - \lambda_N^2}, \quad \pi_{i,k,\ell} = \frac{\max\limits_{i+n_b\le j\le N}\prod\limits_{m=1}^{k-1}|\sigma_m^2 - \lambda_j^2|}{\min\limits_{k\le t\le\ell}\prod\limits_{m=1}^{k-1}|\sigma_m^2 - \lambda_t^2|},$$

and

$$\|\sin\Theta_K(\check{\Xi}_2, \mathcal{Y}_n\Psi_{(:,k:\ell)})\|_F \le \frac{\pi_{i,k}\sqrt{1+c^2\|A_n^T B_n\|_2^2/\delta^2}}{C_{n-i}(1+2\gamma_{i,\ell})}\|\tan\Theta_K(\check{\Xi}_2, Z)\|_F \tag{8}$$

with constant c lies between 1 and $\pi/2$, and $c = 1$ if $k = \ell$, and

$$\delta = \min_{\substack{k\le j\le\ell \\ p<k \text{ or } p>\ell}} |\lambda_j^2 - \sigma_p^2|, \quad \pi_{i,k} = \prod_{j=1}^{i-1}\frac{\lambda_j^2 - \lambda_N^2}{\lambda_j^2 - \lambda_k^2}.$$

Particularly if $\sigma_{k-1}^2 \ge \lambda_k^2$, then

$$\pi_{i,k,\ell} = \prod_{m=1}^{k-1}\frac{|\sigma_m^2 - \lambda_N^2|}{|\sigma_m^2 - \lambda_k^2|}.$$

Proof. Multiplying L^T from left, (4) can be rewritten as $L^T ML(L^T\Xi) = (L^T\Xi)\Lambda^2$, so, $(\lambda_j^2, L^T\xi_j)$ is the eigenpair of $L^T ML$, for $j = 1, \cdots, N$, and $L^T\xi_1, \cdots, L^T\xi_N$ are orthonormal. Do the same process to (3), we have

$$L^T ML\mathcal{V}_n = \mathcal{V}_n\mathcal{B}_n^T\mathcal{B}_n + V_{n+1}B_n^T A_n E_n^T, \tag{9}$$

where $\mathcal{V}_n = L^T\mathcal{Y}_n$, $V_{n+1} = L^T Y_{n+1}$, and $\mathcal{V}_n^T\mathcal{V}_n = I_{nn_b}$, which can be seen as the relation generalize by using standard Lanczos process to $L^T ML$. Thus, $\sigma_1^2, \cdots, \sigma_{nn_b}^2$ are the Ritz values of $L^T ML$, with the corresponding orthonormal Ritz vectors $\mathcal{V}_n\psi_1, \cdots, \mathcal{V}_n\psi_{nn_b}$.

Premultiplying L^T to Equation (6), we have $L^T\Xi_2\Xi_2^T L(L^T Z) = L^T\check{\Xi}_2$. Consequently, the conditions of the block Lanczos convergence Theorem 4.1 and Theorem 5.1 in [17] are satisfied. Thus, using the results Theorem 5.1 in [17], one has

$$\|diag(\lambda_k^2 - \sigma_k^2, \cdots, \lambda_\ell^2 - \sigma_\ell^2)\|_F \le (\lambda_k^2 - \lambda_N^2)\frac{\pi_{i,k,\ell}^2}{C_{n-k}^2(1+2\gamma_{i,\ell})}\|\tan^2\Theta(L^T\check{\Xi}_2, L^T Z)\|_F.$$

Then the bound (7) can be easily got by using ([21] Theorem 4.2)

$$\Theta(L^T\check{\Xi}_2, L^T Z) = \Theta_K(\check{\Xi}_2, Z). \tag{10}$$

Let $\Pi_n = \mathcal{V}_n\mathcal{V}_n^T$, then Π_n is the orthogonal projection onto $\mathcal{K}_n(L^T ML, L^T Z)$, thus from (9), we have

$$\begin{aligned}\|\Pi_n L^T ML(I - \Pi_n)\|_2 &= \|\mathcal{V}_n\mathcal{V}_n^T L^T ML(I - \mathcal{V}_n\mathcal{V}_n^T)\|_2\\ &= \|\mathcal{V}_n(\mathcal{B}_n^T\mathcal{B}_n + E_n A_n^T B_n V_{n+1}^T) - \mathcal{V}_n\mathcal{B}_n^T\mathcal{B}_n\mathcal{V}_n^T\|_2\\ &= \|V_n A_n^T B_n V_{n+1}^T\|_2\\ &= \|A_n^T B_n\|_2.\end{aligned}$$

Consequently, applying the results of Theorem 4.1 in [17], we get

$$\begin{aligned}\|\sin\Theta(L^T\check{\Xi}_2, \mathcal{V}_n\Psi_{(:,k:\ell)})\|_F &\le \frac{\pi_{i,k}\sqrt{1 + \|\Pi_n L^T ML(I - \Pi_n)\|_2^2/\delta^2}}{C_{n-i}(1+2\gamma_{i,\ell})}\|\tan\Theta(L^T\check{\Xi}_2, L^T Z)\|_F\\ &= \frac{\pi_{i,k}\sqrt{1 + \|A_n^T B_n\|_2^2/\delta^2}}{C_{n-i}(1+2\gamma_{i,\ell})}\|\tan\Theta(L^T\check{\Xi}_2, L^T Z)\|_F.\end{aligned}$$

Then the bound (8) can be derived by using $\Theta(L^T\check{\Xi}_2, \mathcal{V}_n\Psi_{(:,k:\ell)}) = \Theta_K(\check{\Xi}_2, \mathcal{Y}_n\Psi_{(:,k:\ell)})$ and (10). □

Theorem 1 is used to bound the errors of the approximate eigenvalues to an eigenvalue cluster including the multiple eigenvalues. It can be also applied to the single eigenvalue case, the following corollary is derived by setting $k = \ell = i$, except the left equality of (10), which needs to be proved.

Corollary 1. *Suppose $\theta_K^{(1)}(Y_1, \Xi_2) < \pi/2$, then for $1 \le i \le nn_b$, there exits a vector $y \in \mathcal{R}(Y_1)$, s.t., $\Xi_2\Xi_2^T y = \xi_i$, and*

$$\lambda_i^2 - \sigma_i^2 \le (\lambda_i^2 - \lambda_N^2)\frac{\pi_{i,j}^2}{C_{n-i}^2(1+2\gamma_i)}\tan^2\theta_K(\xi_i, y)$$

with

$$\gamma_i = \frac{\lambda_i^2 - \lambda_{i+n_b}^2}{\lambda_{i+n_b}^2 - \lambda_N^2}, \quad \pi_{i,j} = \max_{i+n_b \le j \le N}\prod_{m=1}^{i-1}\frac{|\sigma_m^2 - \lambda_j^2|}{|\sigma_m^2 - \lambda_i^2|},$$

and

$$\left((1 - \frac{\sigma_i^2}{\lambda_i^2}) + \frac{\sigma_i^2}{\lambda_i^2} \sin^2 \theta_M(\eta_i, \mathcal{X}_n \phi_i) \right)^{1/2}$$
$$= \sin \theta_K(\xi_i, \mathcal{Y}_n \psi_i) \leq \frac{\pi_i \sqrt{1 + \|A_n^T B_n\|_2^2 / \delta^2}}{C_{n-i}(1 + 2\gamma_i)} \tan \theta_K(\xi_i, y) \tag{11}$$

with

$$\delta = \min_{i \neq j} |\lambda_j^2 - \sigma_i^2|, \quad \pi_i = \prod_{j=1}^{i-1} \frac{\lambda_j^2 - \lambda_N^2}{\lambda_j^2 - \lambda_i^2}.$$

Proof. We only proof the left equality of (11). From (4) and Lemma 2, we have $\Xi = MK\Xi\Lambda^{-2} = M\Gamma\Lambda^{-1}$. If we let $Z_1 = (\mathcal{Y}_n \psi_i)^T K \xi_i$, and $Z_2 = (\mathcal{X}_n \phi_i)^T M \eta_i$, then we can get $Z_1 = \frac{\sigma_i}{\lambda_i} Z_2$ by using $K\mathcal{Y}_n \Psi = \mathcal{X}_n B_n \Psi = \mathcal{X}_n \Phi \Sigma_n$. Thus

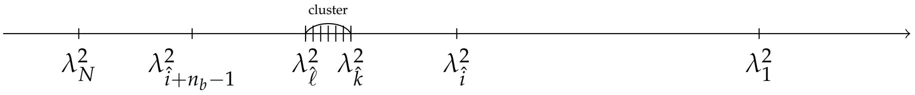
$$\begin{aligned} \sin^2 \theta_K(\xi_i, \mathcal{Y}_n \psi_i) &= 1 - \cos^2 \theta_K(\xi_i, \mathcal{Y}_n \psi_i) \\ &= 1 - Z_1^T Z_1 \\ &= 1 - \frac{\sigma_i^2}{\lambda_i^2} Z_2^T Z_2 \\ &= 1 - \frac{\sigma_i^2}{\lambda_i^2} \cos^2 \theta_M(\eta_i, \mathcal{X}_n \phi_i) \\ &= 1 - \frac{\sigma_i^2}{\lambda_i^2} + \frac{\sigma_i^2}{\lambda_i^2} \sin^2 \theta_M(\eta_i, \mathcal{X}_n \phi_i). \end{aligned}$$

Then,

$$\sin \theta_K(\xi_i, \mathcal{Y}_n \psi_i) = \left(1 - \frac{\sigma_i^2}{\lambda_i^2} + \frac{\sigma_i^2}{\lambda_i^2} \sin^2 \theta_M(\eta_i, \mathcal{X}_n \phi_i) \right)^{1/2}.$$

□

Next, we are going to consider the last few σ_j to approximate as the last few λ_{N-nn_b+j}, $j = k, \cdots, \ell$, and λ_{N-nn_b+k} to $\lambda_{N-nn_b+\ell}$ form a cluster in $\lambda_{\hat{i}}$ to $\lambda_{\hat{i}+n_b-1}$, which is described in the following figure, where $N + 1 - nn_b \leq \hat{i} \leq \hat{k} \leq \hat{\ell} \leq \hat{i} + n_b - 1 \leq N$, $nn_b - \ell + 1 \leq n$, $\hat{k} \triangleq N - nn_b + k$, and $\hat{\ell} \triangleq N - nn_b + \ell$.



Similar to the above discussion for the first few eigenvalues, we can also obtain the error bounds of the approximate last few eigenpairs belongs to eigenvalue cluster together. We use the same notion, except $\hat{\Lambda}_2^2 = diag(\lambda_{\hat{k}}^2, \cdots, \lambda_{\hat{\ell}}^2)$ and $\hat{\Xi}_2 = \Xi_{(:,\hat{k}:\hat{\ell})}$. Assuming $\theta_K^{(1)}(Y_1, \Xi_2) < \pi/2$, then from Lemma 1, there $\exists\ \hat{Z} \in \mathbb{R}^{N \times (\ell - k + 1)}$ with $\mathcal{R}(\hat{Z}) \subseteq \mathcal{R}(Y_1)$, s.t.,

$$\Xi_2 \Xi_2^T K \hat{Z} = \hat{\Xi}_2. \tag{12}$$

Theorem 2. *Suppose $\theta_K^{(1)}(\Upsilon_1, \Xi_2) < \pi/2$ and $\hat{Z}$ satisfy (12), then we have*

$$\begin{aligned} &\|diag(\sigma_k^2 - \lambda_{\hat{k}}^2, \cdots, \sigma_\ell^2 - \lambda_{\hat{\ell}}^2)\|_F \\ \leq &(\lambda_1^2 - \lambda_{\hat{\ell}}^2) \frac{\hat{\pi}_{\hat{i},\hat{k},\hat{\ell}}^2}{C_{n-N+\hat{\ell}-1}^2(1+2\hat{\gamma}_{\hat{i},\hat{k}})} \|\tan^2\Theta_K(\hat{\Xi}_2, \hat{Z})\|_F \end{aligned} \tag{13}$$

with

$$\hat{\gamma}_{\hat{i},\hat{\ell}} = \frac{\lambda_{\hat{i}-1}^2 - \lambda_{\hat{k}}^2}{\lambda_1^2 - \lambda_{\hat{i}-1}^2}, \quad \hat{\pi}_{\hat{i},\hat{k},\hat{\ell}} = \frac{\max\limits_{1 \leq j \leq \hat{i}-1} \prod\limits_{m=\ell+1}^{nn_b} |\sigma_m^2 - \lambda_j^2|}{\min\limits_{\hat{k} \leq t \leq \hat{\ell}} \prod\limits_{m=\ell+1}^{nn_b} |\sigma_m^2 - \lambda_t^2|},$$

and

$$\|\sin\Theta_K(\hat{\Xi}_2, \mathcal{Y}_n\Psi_{(:,k:\ell)})\|_F \leq \frac{\hat{\pi}_{\hat{i},\hat{\ell}}\sqrt{1 + \hat{c}^2\|A_n^T B_n\|_2^2/\hat{\delta}^2}}{C_{n+\hat{i}+n_b-N-2}(1+2\hat{\gamma}_{\hat{i},\hat{k}})} \|\tan\Theta_K(\hat{\Xi}_2, \hat{Z})\|_F \tag{14}$$

with constant $\hat{c}$ lies between 1 and $\pi/2$, and $\hat{c} = 1$ if $k = \ell$, and

$$\hat{\delta} = \min_{\substack{\hat{k} \leq j \leq \hat{\ell} \\ p < k \text{ or } p > \ell}} |\lambda_j^2 - \sigma_p^2|, \quad \hat{\pi}_{\hat{i},\hat{\ell}} = \prod_{j=\hat{i}+n_b}^{N} \frac{\lambda_1^2 - \lambda_j^2}{\lambda_{\hat{\ell}}^2 - \lambda_j^2}.$$

Remark 3. *Similar to Corollary 1, Theorem 2 can also be applied to the single eigenvalue case, here we omit the detail.*

Remark 4. *In Theorem 1 and 2, we use the Frobenius norm to estimate the accuracy of eigenpairs approximations, in fact, any unitary invariant norm can be used to measure.*

Remark 5. *Compared with the single-vector type of the weighted Golub-Kahan-Lanczos method in [16], our convergence results show the superiority of the block version. For instance, in Corollary 1, the convergence rate of the approximate eigenvalues σ_j is proportional to $C_{n-i}^{-2}(1+2\gamma_i)$ with $\gamma_i = \frac{\lambda_i^2 - \lambda_{i+n_b}^2}{\lambda_{i+n_b}^2 - \lambda_N^2}$, which is obviously better than $C_{n-i}^{-2}(1+2\tilde{\gamma}_i)$ with $\tilde{\gamma}_i = \frac{\lambda_i^2 - \lambda_{i+1}^2}{\lambda_{i+1}^2 - \lambda_N^2}$ in ([16] Theorem 3.4). While the additional cost caused from the block version can be paid by the improvements generated by γ_i, especially when the desired eigenvalues lie in a well-separated cluster [12].*

4. Thick Restart

As the number of iterations increases, Algorithm 1 may encounter the dilemma that the amount of calculation and storage increases sharply and the numerical stability gradually weakens. In this section, we will apply the thick restart strategy [20] to improve the algorithm. After running n iterations, Algorithm 1 derives the following relations for LREP:

$$\begin{cases} K\mathcal{Y}_n = \mathcal{X}_n\mathcal{B}_n, \\ M\mathcal{X}_n = \mathcal{Y}_n\mathcal{B}_n^T + Y_{n+1}B_n^T E_n^T, \end{cases} \tag{15}$$

with $\mathcal{X}_n^T M\mathcal{X}_n = I_{nn_b} = \mathcal{Y}_n^T K\mathcal{Y}_n$.

Recall the SVD (2), let Φ_k and Ψ_k be the first kn_b columns of Φ and Ψ, respectively, i.e.,

$$\Phi_k = [\phi_1, \phi_2, \cdots, \phi_{kn_b}], \quad \Psi_k = [\psi_1, \psi_2, \cdots, \psi_{kn_b}].$$

Thus it follows that

$$\mathcal{B}_n\Psi_k = \Phi_k\Sigma_k \quad \text{and} \quad \mathcal{B}_n^T\Phi_k = \Psi_k\Sigma_k, \tag{16}$$

where $\Sigma_k = diag(\sigma_1, \cdots, \sigma_{kn_b})$.

By using the approximate eigenvectors of $\mathbf{H}$ for thick restart, we post-multiply Ψ_k and Φ_k to the Equation (15), respectively, and get

$$\begin{cases} K\mathcal{Y}_n\Psi_k = \mathcal{X}_n\mathcal{B}_n\Psi_k, \\ M\mathcal{X}_n\Phi_k = \mathcal{Y}_n\mathcal{B}_n^T\Phi_k + Y_{n+1}B_n^T E_n^T\Phi_k, \end{cases} \tag{17}$$

From (16), and let $\widehat{\mathcal{Y}}_k = \mathcal{Y}_n\Psi_k$, $\widehat{\mathcal{X}}_k = \mathcal{X}_n\Phi_k$, $\widehat{\mathcal{B}}_k = \Sigma_k$, $\widehat{Y}_{k+1} = Y_{n+1}$, $U^T = E_n^T\Phi_k$, $\widehat{B}_k = B_n$, then (17) can be rewritten as

$$\begin{cases} K\widehat{\mathcal{Y}}_k = \widehat{\mathcal{X}}_k\widehat{\mathcal{B}}_k, \\ M\widehat{\mathcal{X}}_k = \widehat{\mathcal{Y}}_k\widehat{\mathcal{B}}_k^T + \widehat{Y}_{k+1}\widehat{B}_k^T U^T, \end{cases} \tag{18}$$

and $\widehat{\mathcal{X}}_k^T M\widehat{\mathcal{X}}_k = I_{kn_b} = \widehat{\mathcal{Y}}_k^T K\widehat{\mathcal{Y}}_k$.

Next, $\widehat{X}_{k+1}$ and $\widehat{Y}_{k+2}$ will be generalized. Firstly, we compute

$$\begin{aligned} \widetilde{X}_{k+1} &= K\widehat{Y}_{k+1} - \widehat{\mathcal{X}}_k\widehat{\mathcal{X}}_k^T MK\widehat{Y}_{k+1} \\ &= K\widehat{Y}_{k+1} - \widehat{\mathcal{X}}_k U\widehat{B}_k. \end{aligned}$$

From the second equation in (18), we know $\widetilde{X}_{k+1}^T M\widehat{\mathcal{X}}_k = 0$. Do Cholesky decomposition $\widetilde{X}_{k+1}^T M\widetilde{X}_{k+1} = W^TW$, and set $\widehat{A}_{k+1} = W$, $W = inv(W)$. Compute $\widehat{X}_{k+1} = \widetilde{X}_{k+1}W$, and let

$$\widehat{\mathcal{X}}_{k+1} = [\widehat{\mathcal{X}}_k, \widehat{X}_{k+1}], \quad \widehat{\mathcal{B}}_{k+1} = \begin{bmatrix} \widehat{\mathcal{B}}_k & U\widehat{B}_k \\ 0 & \widehat{A}_{k+1} \end{bmatrix},$$

we have

$$K\widehat{\mathcal{Y}}_{k+1} = \widehat{\mathcal{X}}_{k+1}\widehat{\mathcal{B}}_{k+1} \quad \text{with} \quad \widehat{\mathcal{X}}_{k+1}^T M\widehat{\mathcal{X}}_{k+1} = I_{(k+1)n_b}. \tag{19}$$

Secondly, from the above equation, we can compute

$$\begin{aligned} \widetilde{Y}_{k+2} &= M\widehat{X}_{k+1} - \widehat{\mathcal{Y}}_k\widehat{\mathcal{Y}}_k^T KM\widehat{X}_{k+1} - \widehat{Y}_{k+1}\widehat{Y}_{k+1}^T KM\widehat{X}_{k+1} \\ &= M\widehat{X}_{k+1} - \widehat{Y}_{k+1}\widehat{A}_{k+1}^T. \end{aligned}$$

Again using (19), it is easily got that $\widetilde{Y}_{k+2}^T K\widehat{\mathcal{Y}}_{k+1} = 0$. Similarly, do Cholesky decomposition $\widetilde{Y}_{k+2}^T K\widetilde{Y}_{k+2} = W^TW$, and let $\widehat{B}_{k+1} = W^T$, $W = inv(W)$. Compute $\widehat{Y}_{k+2} = \widetilde{Y}_{k+2}W$, and let $\widehat{\mathcal{Y}}_{k+1} = [\widehat{\mathcal{Y}}_k, \widehat{Y}_{k+1}]$, we get

$$M\widehat{\mathcal{X}}_{k+1} = \widehat{\mathcal{Y}}_{k+1}\widehat{\mathcal{B}}_{k+1}^T + \widehat{Y}_{k+2}\widehat{B}_{k+1}^T E_{k+1}^T \quad \text{with} \quad \widehat{\mathcal{Y}}_{k+1}^T M\widehat{\mathcal{Y}}_{k+1} = I_{(k+1)n_b}.$$

Continue the same procedure for $\widehat{X}_{k+2}, \cdots, \widehat{X}_n$ and $\widehat{Y}_{k+3}, \cdots, \widehat{Y}_{n+1}$, we can obtain the new M-orthonormal matrix $\widehat{\mathcal{X}}_n \in \mathbb{R}^{N\times nn_b}$, the new K-orthonormal matrix $\widehat{\mathcal{Y}}_n \in \mathbb{R}^{N\times nn_b}$, and the new matrix $\widehat{\mathcal{B}}_n \in \mathbb{R}^{nn_b\times nn_b}$, and relations

$$\begin{cases} K\widehat{\mathcal{Y}}_n = \widehat{\mathcal{X}}_n\widehat{\mathcal{B}}_n, \\ M\widehat{\mathcal{X}}_n = \widehat{\mathcal{Y}}_n\widehat{\mathcal{B}}_n^T + \widehat{Y}_{n+1}\widehat{B}_n^T E_n^T, \end{cases} \tag{20}$$

with $\widehat{\mathcal{X}}_n^T M\widehat{\mathcal{X}}_n = I_{nn_b} = \widehat{\mathcal{Y}}_n^T K\widehat{\mathcal{Y}}_n$, and

$$\widehat{\mathcal{B}}_n = \begin{bmatrix} \widehat{\mathcal{B}}_k & U\widehat{B}_k & & \\ & \widehat{A}_{k+1} & \widehat{B}_{k+1} & \\ & & \ddots & \widehat{B}_{n-1} \\ & & & \widehat{A}_n \end{bmatrix}.$$

Note that $\widehat{\mathcal{B}}_n$ is no longer a block bidiagonal matrix. Algorithm 2 is our thick-restart weighted block Golub-Kahan-Lanczos algorithm for LREP.

Remark 6. *Actually, from the construction of $\widehat{\mathcal{B}}_n$, we can know the procedure for getting $\widehat{X}_{k+2}, \cdots, \widehat{X}_n$ and $\widehat{Y}_{k+3}, \cdots, \widehat{Y}_{n+1}$ is the same as applying Algorithm 1 to $\widehat{Y}_{k+2}$ for $n-k-1$ iterations, thus we use Algorithm 1 directly in restarting* **Step 2** *of the following Algorithm 2.*

Algorithm 2: wbGKL-TR

```
1. Given an initial guess $Y_1$ satisfying $Y_1^T K Y_1 = I_{n_b}$, a tolerance $tol$, an integer $k$ that the $k$ blocks approximate eigenvectors we want to add to the solving subspace, an integer $n$ the block dimension of solving subspace, as well as $w_\ell$ the desired number of eigenpairs;
2. Apply Algorithm 1 from the current point to generate the rest of $\mathcal{X}_n$, $\mathcal{Y}_{n+1}$, and $\mathcal{B}_n$. If it is the first cycle, the current point is $Y_1$, else $Y_{k+2}$;
3. Compute an SVD of $\mathcal{B}_n$ as in (2), select $w_\ell (w_\ell \leq n n_b)$ wanted singular values $\sigma_j$, and their associated left singular vectors $\phi_j$ and right singular vectors $\psi_j$. Form the approximate eigenpairs for H, if the stopping criterion is satisfied, then stop, else continue;
4. Generate new $\widehat{\mathcal{X}}_{k+1}$, $\widehat{\mathcal{Y}}_{k+2}$ and $\widehat{\mathcal{B}}_{k+1}$:
Compute $\widehat{\mathcal{Y}}_k = \mathcal{Y}_n \Psi_k$, $\widehat{\mathcal{X}}_k = \mathcal{X}_n \Phi_k$, $\widehat{\mathcal{B}}_k = \Sigma_k$, $\widehat{Y}_{k+1} = Y_{n+1}$, $U^T = E_n^T \Phi_k$, $\widehat{B}_k = B_n$;
Compute $\widetilde{X}_{k+1} = K\widehat{Y}_{k+1} - \widehat{\mathcal{X}}_k U \widehat{B}_k$, do Cholesky decomposition $\widetilde{X}_{k+1}^T M \widetilde{X}_{k+1} = W^T W$, set
 $\widehat{A}_{k+1} = W$, $W = inv(W)$, $\widehat{X}_{k+1} = \widetilde{X}_{k+1} W$;
Compute $\widetilde{Y}_{k+2} = M\widehat{X}_{k+1} - \widehat{Y}_{k+1} \widehat{A}_{k+1}^T$, do Cholesky decomposition $\widetilde{Y}_{k+2}^T K \widetilde{Y}_{k+2} = W^T W$, set
 $\widehat{B}_{k+1} = W^T$, $W = inv(W)$, $\widehat{Y}_{k+2} = \widetilde{Y}_{k+2} W$;
Let $\mathcal{X}_{k+1} = \widehat{\mathcal{X}}_{k+1} = [\widehat{\mathcal{X}}_k, \widehat{X}_{k+1}]$, $\mathcal{B}_{k+1} = \widehat{\mathcal{B}}_{k+1} = \begin{bmatrix} \widehat{\mathcal{B}}_k & U\widehat{B}_k \\ 0 & \widehat{A}_{k+1} \end{bmatrix}$, $\mathcal{Y}_{k+2} = \widehat{\mathcal{Y}}_{k+2} = [\widehat{\mathcal{Y}}_k, \widehat{Y}_{k+1}, \widehat{Y}_{k+2}]$,
 and go to Step 2.
```

Remark 7. *In* **Step 3**, *we compute the harmonic Ritz pairs after n iterations. In practice, we do the computation for each iterations $j = 1, \cdots, n$. When restarting, the information chosen to add to the solving subspaces are the wanted w_ℓ singular values of $\mathcal{B}_n$ with their corresponding left and right singular vectors. Actually, we use MATLAB command "sort" to choose the w_ℓ smallest ones or the w_ℓ largest ones, and which singular values to choose depends on the desired eigenvalues of* **H**.

In the end of this section, we list the computational costs in a generic cycle of four algorithms, which are weighted block Golub-Kahan-Lanczos algorithm, thick-restart weighted block Golub-Kahan-Lanczos algorithm, block Lanczos algorithm [12], and thick-restart block Lanczos algorithm [12], and denoted by **wbGKL**, **wbGKL-TR**, **BLan**, and **BLan-TR**, respectively. The detail pseudocodes of **BLan** and **BLan-TR** are be found in [12].

The comparisons are presented in Tables 1 and 2. Here, we denote "block vector" a $N \times n_b$ rectangular matrix, denote "mvb" the product number of a $N \times N$ matrix and a block vector. "dpb" denotes the dot product number of two block vectors X and Y, i.e., $X^T Y$. "saxpyb" denotes the number of adding two block vectors or multiplying a block vector to a $n_b \times n_b$ small matrix. "Ep($2n \times 2n$)(with sorting)" means the number of $2n \times 2n$ size eigenvalue problem with sorting eigenvalues and their corresponding eigenvectors in one cycle. Similarly, "Sp($n \times n$)" denotes the number of $n \times n$ size singular value decomposition in one cycle. Because **wbGKL** and **BLan** are non-restart algorithms, we just count the first n Lanczos iterations.

Table 1. Main computational costs per cycle **wbGKL** and **wbGKL-TR.**

	wbGKL	**wbGKL-TR (1-st Cycle)**	**wbGKL-TR (Other Cycle)**
mvb	$2n+1$	$2n+1$	$2(n-k)$
dpb	$2n+1$	$2n+1$	$2(n-k)$
saxpyb	$8n$	$8n$	$8(n-k)+2k(2n+1)$
block vector updates	$2n+2$	$2n+2$	$2n+2$
Ep($2n \times 2n$)(with sorting)	0	0	0
Sp($n \times n$)	1	1	1

Table 2. Main computational costs per cycle **BLan** and **BLan-TR.**

	BLan	**BLan-TR (1-st Cycle)**	**BLan-TR (Other Cycle)**
mvb	$2n+1$	$2n+1$	$2(n-k)$
dpb	$2n+1$	$2n+1$	$2(n-k)$
saxpyb	$6n$	$6n$	$6(n-k)+2k(2n+1)$
block vector updates	$2n+2$	$2n+2$	$2n+2$
Ep($2n \times 2n$)(with sorting)	1	1	1
Sp($n \times n$)	0	0	0

5. Numerical Examples

In this section, two numerical experiments are carried out by using MATLAB 8.4 (R2014b) on a laptop with an Intel Core i5-6200U CPU 2.3 GHz memory 8 GB under the Windows 10 operating system.

Example 1. *In this example, we check the bounds established in Theorem 1 and 2. For simplicity, we take $N = 100$, the number of weighted block Golub-Kahan-Lanczos steps $n = 20$, $K = M$ as diagonal matrix $diag(\lambda_1, \lambda_2, \cdots, \lambda_N)$, where*

$$\lambda_1 = 11+\rho,\ \ \lambda_2 = 11,\ \ \lambda_3 = 11-\rho,$$
$$\lambda_{N-2} = 1+\rho,\ \ \lambda_{N-1} = 1,\ \ \lambda_N = 1-\rho,$$
$$\lambda_j = 5 + \frac{5(N-j+1)}{N-3},\ \ j = 4, \cdots, N-3,$$

and $i = k = 1$, $\ell = 3$, $\hat{i} = \hat{k} = N-2$, $\hat{\ell} = N$, $n_b = 3$. There are three positive eigenvalue clusters: $\{\lambda_1, \lambda_2, \lambda_3\}$, $\{\lambda_4, \cdots, \lambda_{N-3}\}$, or $\{\lambda_{N-2}, \lambda_{N-1}, \lambda_N\}$. Obviously, $\Xi = \Gamma = K^{-\frac{1}{2}}$.

We seek two groups of the approximate eigenpairs, the first is related to the first cluster, the second is related to the last cluster, i.e., $\{\sigma_1, \sigma_2, \sigma_3\}$ approximate $\{\lambda_1, \lambda_2, \lambda_3\}$, and $\{\sigma_{nn_b-2}, \sigma_{nn_b-1}, \sigma_{nn_b}\}$ approximate $\{\lambda_{N-2}, \lambda_{N-1}, \lambda_N\}$. In order to see the affect that generated from ρ to the upper bounds of the approximate eigenpairs errors in weighted block Golub-Kahan-Lanczos method for LREP, we change the parameter $\rho > 0$ to overmaster the tightness among eigenvalues within $\{\lambda_1, \lambda_2, \lambda_3\}$ and $\{\lambda_{N-2}, \lambda_{N-1}, \lambda_N\}$. First, we choose the same matrix Y_0 as in [12,17], i.e.,

$$Y_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ \frac{1}{N} & sin1 & cos1 \\ \vdots & \vdots & \vdots \\ \frac{N-n_b}{N} & sin(N-n_b) & cos(N-n_b) \end{bmatrix}.$$

Obviously, $rank(Y_0) = n_b$ and $rank(Y_0^T K \Xi_{(:,1:3)}) = n_b$. Since K symmetric positive definite, thus do Cholesky decomposition $Y_0^T K Y_0 = W^T W$, let $Y_1 = Y_0 W^{-1}$, hence, Y_1 satisfies (5), i.e., $Y_1^T K \Xi_{(:,1:3)}$ is singular. We take $Z = Y_1(\Xi_{(:,1:3)}^T K Y_1)^{-1}$, then Z satisfies (6). We execute the weighted block Golub-Kahan-Lanczos method with full re-orthogonalization for LREP in MATLAB, and check the bounds in (7), (8), (13), and (14). Since the approximate eigenvalues are $\{\sigma_1, \sigma_2, \sigma_3\}$ and $\{\sigma_{nn_b-2}, \sigma_{nn_b-1}, \sigma_{nn_b}\}$, thus $\pi_{i,k,\ell} = \pi_{i,k} = \hat{\pi}_{\hat{i},\hat{k},\hat{\ell}} = \hat{\pi}_{\hat{i},\hat{\ell}} = 1$, $c = \hat{c} = 1$, and we measure the following two groups of errors:

$$
\begin{aligned}
\varepsilon_{11} &= \|diag(\lambda_1^2 - \sigma_1^2, \lambda_2^2 - \sigma_2^2, \lambda_3^2 - \sigma_3^2)\|_F,\\
\varepsilon_{21} &= \frac{\lambda_1^2 - \lambda_N^2}{C_{n-1}^2(1 + 2\gamma_{1,3})} \|\tan^2 \Theta_K(\Xi_{(:,1:3)}, Z)\|_F,\\
\varepsilon_{31} &= \|\sin \Theta_K(\Xi_{(:,1:3)}, \mathcal{Y}_n \Psi_{(:,1:3)})\|_F,\\
\varepsilon_{41} &= \frac{\sqrt{1 + \|A_n^T B_n\|_2^2/\delta^2}}{C_{n-1}(1 + 2\gamma_{1,3})} \|\tan \Theta_K(\Xi_{(:,1:3)}, Z)\|_F,
\end{aligned}
$$

and

$$
\begin{aligned}
\varepsilon_{12} &= \|diag(\sigma_{N-2}^2 - \lambda_{N-2}^2, \sigma_{N-1}^2 - \lambda_{N-1}^2, \sigma_N^2 - \lambda_N^2)\|_F,\\
\varepsilon_{22} &= \frac{\lambda_1^2 - \lambda_N^2}{C_{n-1}^2(1 + 2\hat{\gamma}_{N-2,N})} \|\tan^2 \Theta_K(\Xi_{(:,N-2:N)}, Z)\|_F,\\
\varepsilon_{32} &= \|\sin \Theta_K(\Xi_{(:,N-2:N)}, \mathcal{Y}_n \Psi_{(:,nn_b-2:nn_b)})\|_F,\\
\varepsilon_{42} &= \frac{\sqrt{1 + \|A_n^T B_n\|_2^2/\hat{\delta}^2}}{C_{n-i}(1 + 2\hat{\gamma}_{N-2,N})} \|\tan \Theta_K(\Xi_{(:,N-2:N)}, Z)\|_F.
\end{aligned}
$$

Actually, ε_{21} and ε_{41} are upper bounds of ε_{11} and ε_{31}, and ε_{22} and ε_{42} are upper bounds of ε_{12} and ε_{32}. Tables 3 and 4 report the results of ε_{ij}, $i = 1, \cdots, 4$, $j = 1, 2$ with the parameter ρ goes to 0. From the two tables, we can see that the bounds for the eigenvalues lie in a cluster and their corresponding eigenvectors are sharp, and they are not sensitive to ρ when ρ goes to 0.

Table 3. ε_{11}, ε_{31} together with their upper bounds ε_{21}, ε_{41} of Example 1.

ρ	ε_{11}	ε_{21}	ε_{31}	ε_{41}
10^{-1}	4.0295×10^{-13}	2.6773×10^{-10}	1.2491×10^{-10}	2.6260×10^{-6}
10^{-2}	5.1238×10^{-14}	5.4555×10^{-11}	6.1184×10^{-11}	1.1407×10^{-6}
10^{-3}	7.1054×10^{-14}	4.6711×10^{-11}	5.7698×10^{-11}	1.0520×10^{-6}
10^{-4}	2.4449×10^{-13}	4.5993×10^{-11}	5.7370×10^{-11}	1.0436×10^{-6}
10^{-5}	2.1552×10^{-13}	4.5922×10^{-11}	5.7338×10^{-11}	1.0427×10^{-6}

Table 4. ε_{12}, ε_{32} together with their upper bounds ε_{22}, ε_{42} of Example 1.

ρ	ε_{11}	ε_{21}	ε_{31}	ε_{41}
10^{-1}	7.1089×10^{-16}	6.0352×10^{-11}	1.9393×10^{-10}	8.8823×10^{-7}
10^{-2}	1.3688×10^{-15}	3.5913×10^{-11}	1.9562×10^{-10}	6.8797×10^{-7}
10^{-3}	3.9968×10^{-15}	3.4113×10^{-11}	1.9580×10^{-10}	6.7081×10^{-7}
10^{-4}	4.8495×10^{-15}	3.3938×10^{-11}	1.9582×10^{-10}	6.6912×10^{-7}
10^{-5}	8.1221×10^{-15}	3.3920×10^{-11}	1.9582×10^{-10}	6.6895×10^{-7}

Example 2. *In this example, we are going to test the effectiveness of our weighted block Golub-Kahan-Lanczos algorithms. Four algorithms are tested, i.e.,* **wbGKL**, **wbGKL-TR**, **BLan**, *and* **BLan-TR**. *We choose 3 test problems used in [12,13], which are listed in Table 5. All the matrices K and M in the problems are symmetric positive definite. Specifically, Test 1 and Test 2, which are derived by the turboTDDFT command in QUANTUM*

ESPRESSO [22], are from the linear response research for Na2 and silane (SiH4) compound, respectively. The matrices K and M in Test 3 are from the University of Florida Sparse Matrix Collection [23], where the order of K is $N = 9604$*, and M is the leading* $N \times N$ *principal submatrix of* $finan512$.

Table 5. The matrices K and M in Test 1–3.

Problems	N	K	M
Test 1	1862	$Na2$	$Na2$
Test 2	5660	$SiH4$	$SiH4$
Test 3	9604	$fv1$	$finan512$

We aim to compute the smallest 5 positive eigenvalues and the largest 5 eigenvalues, i.e., λ_i for $i = 1, \cdots, 5, N-4, \cdots, N$, together with their associated eigenvectors. The initial guess is chosen as $V_0 = eye(N, n_b)$ with block size $n_b = 3$, where *eye* is the MATLAB command. The same as in Example 1, since K is symmetric positive definite, thus do Cholesky decomposition $Y_0^T K Y_0 = W^T W$, let $Y_1 = Y_0 W^{-1}$, hence, Y_1 satisfies $Y_1^T K Y_1 = I_{n_b}$. In **wbGKL-TR** and **BLan-TR**, we select $n = 30$, $k = 20$, i.e., the restart will occur once the dimension of the solving subspace is larger than 90, and the information of 60 Ritz vectors are kept. For **wbGKL** and **BLan**, because there is no restart, then we compute the approximate eigenpairs when the Lanczos iterations equals to $30 + 10 \times (j-1)$, $j = 1, 2, \cdots$, hence, the Lanczos iterations are as the same amount as in **wbGKL-TR** and **BLan-TR**. The following relative eigenvalue error and relative residual 1-norm for each 10 approximate eigenpairs are calculated:

$$e(\sigma_j) := \begin{cases} \frac{|\lambda_j - \sigma_j|}{\lambda_j}, & j = 1, \cdots, 5, \\ \frac{|\lambda_{n+j-k} - \sigma_j|}{\lambda_{n+j-k}}, & j = nn_b - 4, \cdots, nn_b, \end{cases}$$

$$r(\sigma_j) := \frac{\|\mathbf{H}\tilde{z}_j - \sigma_j \tilde{z}_j\|_1}{(\|\mathbf{H}\|_1 + \sigma_j)\|\tilde{z}_j\|_1}, \quad j = 1, \cdots, 5, nn_b - 4, \cdots, nn_b,$$

where the "exact" eigenvalues λ_j are calculated by the MATLAB code *eig*. The calculated approximate eigenpair $(\sigma_j, \tilde{z}_j)$ is regarded as converged if $r(\sigma_j) \leq tol = 10^{-8}$.

Tables 6 and 7 give the number of the Lanczos iterations (denote by *iter*) and the CPU time in seconds (denote by *CPU*) for the four algorithms, and Table 6 is for the smallest 5 positive eigenvalues, Table 7 is for the largest 5 eigenvalues. From Table 6, one can see that, no matter the smallest or the largest eigenvalues, the iteration number of the four algorithms are competitive, but **wbGKL** and **wbGKL-TR** cost significant less time than **BLan** and **BLan-TR**, especially, **wbGKL-TR** consumes the least amount of time. Because **BLan** and **BLan-TR** need to compute the eigenvalues of $\begin{bmatrix} 0 & T_n \\ D_n & 0 \end{bmatrix}$, which is a nonsymmetric matrix, thus the two algorithms slower than **wbGKL** and **wbGKL-TR**. Due to the saving during the orthogonalization procedure and solving a much smaller $\mathcal{B}_n$, **wbGKL-TR** is the faster algorithm.

Table 6. Compute 5 smallest positive eigenvalues for Test 1–3.

Algorithms	**Test 1**		**Test 2**		**Test 3**	
	CPU	*iter*	*CPU*	*iter*	*CPU*	*iter*
wbGKL	1.5070	149	25.7848	319	15.9308	379
wbGKL-TR	1.0746	179	20.3593	359	5.1302	589
BLan	4.6739	149	87.1670	349	43.9506	379
BLan-TR	2.1243	163	39.1306	393	19.9677	592

Table 7. Compute 5 largest eigenvalues for Test 1–3.

Algorithms	Test 1		Test 2		Test 3	
	CPU	*iter*	*CPU*	*iter*	*CPU*	*iter*
wbGKL	0.6387	79	12.4658	179	1.0639	109
wbGKL-TR	0.5284	79	9.9093	179	0.8774	109
BLan	1.4634	79	27.4028	179	6.7574	109
BLan-TR	1.0151	82	18.3415	186	4.1298	113

The accuracy of the last two approximate eigenpairs in Test 1 are shown in Figure 1. From the figure, we can see that, for the last two eigenpairs, **wbGKL** and **BLan** require almost the same iterations to obtain the same accuracy, and the case of **wbGKL-TR** and **BLan-TR** also need almost the same iterations, which are one or two more restarts than **wbGKL** and **BLan**. On one hand, without solving a nonsymmetric eigenproblem, **wbGKL** and **wbGKL-TR** can save much more time than **BLan** and **BLan-TR**. On the other hand, since the dimension of the solving subspace for **wbGKL-TR** is bounded by nn_b, the savings in the process of orthogonalization and a much smaller singular value decomposition problem is sufficient to cover the additional restart steps.

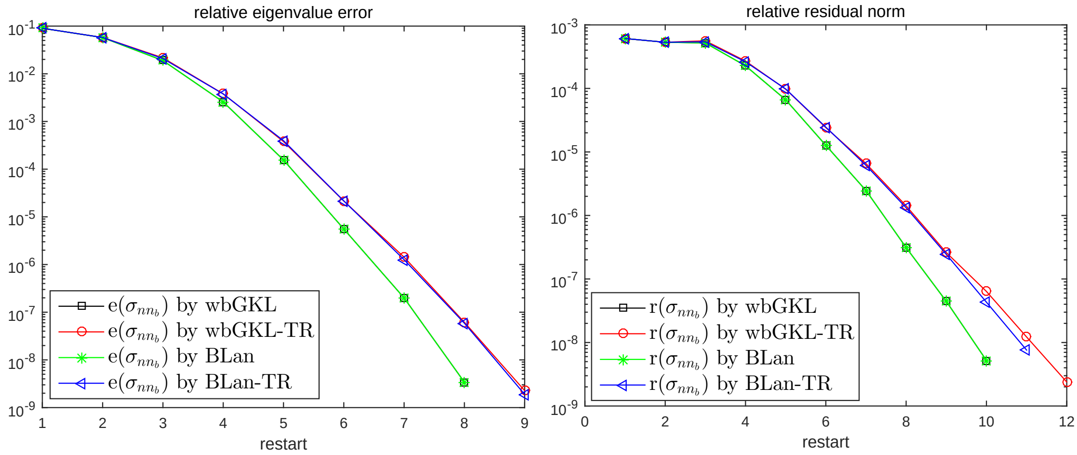


Figure 1. Errors and residuals of the 2 smallest positive eigenvalues for Test 1 in Example 2.

6. Conclusions

In this paper, we present a weighted block Golub-Kahan-Lanczos algorithm to solve the desired small portion of smallest or largest positive eigenvalues which are in a cluster. Convergence analysis is established in Theorems 1 and 2, and bound the errors of the eigenvalue and eigenvector approximations belonging to an eigenvalue cluster. These results also show the advantages of the block algorithm over the single-vector version. To make the new algorithm more practical, we introduced a thick-restart strategy to eliminate the numerical difficulties caused by the block method. Numerical examples are executed to demonstrate the efficiency of our new restart algorithm.

Author Contributions: Conceptualization, G.C.; Data curation, H.Z. and Z.T.; Formal analysis, H.Z. and Z.T.; Methodology, H.Z.; Project administration, H.Z. and G.C.; Resources, H.Z.; Visualization, H.Z. and Z.T.; Writing—original draft, H.Z.; Writing—review and editing, Z.T. and G.C.

Funding: This work was financial supported by the National Nature Science Foundation of China (No. 11701225, 11601081, 11471122), Fundamental Research Funds for the Central Universities (No. JUSRP11719), Natural Science Foundation of Jiangsu Province (No. BK20170173), and the research fund for distinguished young scholars of Fujian Agriculture and Forestry University (No. xjq201727).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Casida, M.E. Time-Dependent Density Functional Response Theory for Molecules. In *Recent Advances in Density Functional Methods*; Chong, D.P., Ed.; World Scientific: Singapore, 1995.
2. Onida, G.; Reining, L.; Rubio, A. Electronic excitations density functional versus many-body Green's function. *Rev. Mod. Phys.* **2002**, *74*, 601–659. [CrossRef]
3. Rocca, D. Time-Dependent Density Functional Perturbation Theory: New algorithms with Applications to Molecular Spectra. Ph.D. Thesis, The International School for Advanced Studies, Trieste, Italy, 2007.
4. Shao, M.; da Jornada, F.H.; Yang, C.; Deslippe, J.; Louie, S.G. Structure preserving parallel algorithms for solving the Bethe-Salpeter eigenvalue problem. *Linear Algebra Appl.* **2016**, *488*, 148–167. [CrossRef]
5. Ring, P.; Ma, Z.; Giai, V.N.; Vretenar, D.; Wandelt, A.; Gao, L. The time-dependent relativistic mean-field theory and the random phase approximation. *Nucl. Phys. A* **2001**, *694*, 249–268. [CrossRef]
6. Bai, Z.; Li, R.-C. Minimization principles for the linear response eigenvalue problem I: Theory. *SIAM J. Matrix Anal. Appl.* **2012**, *33*, 1075–1100. [CrossRef]
7. Bai, Z.; Li, R.-C. Minimization principles for the linear response eigenvalue problem II: Computation. *SIAM J. Matrix Anal. Appl.* **2013**, *34*, 392–416. [CrossRef]
8. Bai, Z.; Li, R.-C. Minimization principles and computation for the generalized linear response eigenvalue problem. *BIT Numer. Math.* **2014**, *54*, 31–54. [CrossRef]
9. Li, T.; Li, R.-C.; Lin, W.-W. A symmetric structure-preserving ΓQR algorithm for linear response eigenvalue problems. *Linear Algebra Appl.* **2017**, *520*, 191–214. [CrossRef]
10. Teng, Z.; Li, R.-C. Convergence analysis of Lanczos-type methods for the linear response eigenvalue problem. *J. Comput. Appl. Math.* **2013**, *247*, 17–33. [CrossRef]
11. Teng, Z.; Lu, L.; Li, R.-C. Perturbation of partitioned linear response eigenvalue problems. *Electron. Trans. Numer. Anal.* **2015**, *44*, 624–638.
12. Teng, Z.; Zhang, L.-H. A block Lanczos method for then linear response eigenvalue problem. *Electron. Trans. Numer. Anal.* **2017**, *46*, 505–523.
13. Teng, Z.; Zhou, Y.; Li, R.-C. A block Chebyshev-Davidson method for linear response eigenvalue problems. *Adv. Comput. Math.* **2016**, *42*, 1103–1128. [CrossRef]
14. Zhang, L.-H.; Lin, W.-W.; Li, R.-C. Backward perturbation analysis and residual-based error bounds for the linear response eigenvalue problem. *BIT Numer. Math.* **2014**, *55*, 869–896. [CrossRef]
15. Zhang, L.-H.; Xue, J.; Li, R.-C. Rayleigh-Ritz approximation for the linear response eigenvalue problem. *SIAM J. Matrix Anal. Appl.* **2014**, *35*, 765–782. [CrossRef]
16. Zhong, H.; Xu, H. Weighted Golub-Kahan-Lanczos bidiagonalizaiton algorithms. *Electron. Trans. Numer. Anal.* **2017**, *47*, 153–178.
17. Li, R.-C.; Zhang, L.-H. Convergence of the block Lanczos method for eigenvalue clusters. *Numer. Math.* **2015**, *131*, 83–113. [CrossRef]
18. Chapman, A.; Saad, Y. Deflated and augmented Krylov subspace techniques. *Numer. Linear Algebra Appl.* **1996**, *4*, 43–66. [CrossRef]
19. Lehoucq, R.B.; Sorensen, D.C. Deflation techniques for an implicitly restarted Arnoldi iteration. *SIAM J. Matrix Anal. Appl.* **1996**, *17*, 789–821. [CrossRef]
20. Wu, K.; Simon, H. Thick-restart Lanczos method for large symmetric eigenvalue problems. *SIAM J. Matrix Anal. Appl.* **2000**, *22*, 602–616. [CrossRef]
21. Knyazev, A.V.; Argentati, M.E. Principal angles between subspaces in an A-based scalar product: Algorithms and perturbation estimates. *SIAM J. Sci. Comput.* **2002**, *23*, 2008–2040. [CrossRef]
22. Giannozzi, P.; Baroni, S.; Bonini, N.; Calandra, M.; Car, R.; Cavazzoni, C.; Ceresoli, D.; Chiarotti, G.L.; Cococcioni, M.; Dabo, I.; et al. QUANTUM ESPRESSO: A modular and open-source software project for quantum simulations of materials. *J. Phys. Condens. Matter* **2009**, *21*, 395502. [CrossRef]
23. Davis, T.; Hu, Y. The University of Florida Sparse Matrix Collection. *ACM Trans. Math. Softw.* **2011**, *38*, 1:1–1:25. [CrossRef]