Linear Algebra and its Applications 435 (2011) 601–622

Contents lists available at ScienceDirect

Linear Algebra and its Applications

journal homepage: www.elsevier.com/locate/laa

LINEAR ALGEBRA and its Applications

Fast inexact subspace iteration for generalized eigenvalue problems with spectral transformation[☆]

Fei Xue [a], Howard C. Elman [b,*]

[a] *Applied Mathematics and Scientific Computation Program, Department of Mathematics, University of Maryland, College Park, MD 20742, USA*
[b] *Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742, USA*

ARTICLE INFO

Submitted by V. Mehrmann

Dedicated to G.W. Stewart on the occasion of his 70th birthday

Keywords:
Inexact subspace iteration
Tuned preconditioner
Deflation
Subspace recycling
Starting vector

ABSTRACT

We study inexact subspace iteration for solving generalized non-Hermitian eigenvalue problems with spectral transformation, with focus on a few strategies that help accelerate preconditioned iterative solution of the linear systems of equations arising in this context. We provide new insights into a special type of preconditioner with “tuning” that has been studied for this algorithm applied to standard eigenvalue problems. Specifically, we propose an alternative way to use the tuned preconditioner to achieve similar performance for generalized problems, and we show that these performance improvements can also be obtained by solving an inexpensive least squares problem. In addition, we show that the cost of iterative solution of the linear systems can be further reduced by using deflation of converged Schur vectors, special starting vectors constructed from previously solved linear systems, and iterative linear solvers with subspace recycling. The effectiveness of these techniques is demonstrated by numerical experiments.

[☆] This work was supported by the U.S. Department of Energy under grant DEFG0204ER25619, and by the U.S. National Science Foundation under grant CCF0726017.

[*] Corresponding author.
E-mail addresses: petrinet@math.umd.edu (F. Xue), elman@cs.umd.edu (H. Elman).

0024-3795/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.laa.2010.06.021

1. Introduction

In many scientific applications, one needs to solve the generalized non-Hermitian eigenvalue problem $Av = \lambda Bv$ for a small group of eigenvalues clustered around a given shift or with largest real parts. The most commonly used approach is to employ a spectral transformation to map these eigenvalues to extremal ones so that they can be easily captured by eigenvalue algorithms. The major difficulty of the spectral transformation is that a linear system of equations involving a shifted matrix $A - \sigma B$ must be solved in each (outer) iteration of the eigenvalue algorithm. If the matrices are so large (for example, those from discretization of three-dimensional partial differential equations) that direct linear solvers based on factorization are too expensive to apply, we have to use iterative methods (inner iteration) to solve these linear systems to some prescribed tolerances. Obviously, the effectiveness of the whole algorithm with "inner–outer" structure depends strongly on that of the inner iteration. In this paper, we study a few techniques that help accelerate the iterative solution of the linear systems that arise when inexact subspace iteration is used to solve the generalized non-Hermitian eigenvalue problem with spectral transformation.

In recent years, great progress has been made in analyzing inexact algorithms, especially inexact inverse iteration, to compute a simple eigenvalue closest to some shift. Refs. [17,20] establish the linear convergence of the outer iteration for non-Hermitian problems, assuming that the algorithm uses a fixed shift and a sequence of decreasing tolerances for the solution of the linear systems. Inexact Rayleigh quotient iteration for symmetric matrices is studied in [32,26], where the authors explore how the inexactness of the solution to the linear systems affects the convergence of the outer iteration. Systematic analysis of this algorithm is given by Spence and his collaborators (see [1–3,12,13]) on the relation between the inner and the outer iterations, with different formulations of the linear systems, and variable shifts and tolerances for solving the linear systems. To make the inner iteration converge more quickly, [31] provides new perspectives on preconditioning by modifying the right hand side of the preconditioned system. This idea is elaborated on in [1–3] and further refined in [12,13] where a special type of preconditioner with "tuning" is defined and shown to greatly reduce the inner iteration counts.

Inexact subspace iteration is a straightforward block extension of inexact inverse iteration with a fixed shift. Robbé et al. [28] establish linear convergence of the outer iteration of this algorithm for standard eigenvalue problems and show by the block-GMRES [30] convergence theory that tuning keeps the block-GMRES iteration counts roughly *constant* for solving the block linear systems, though the inner solve is required to be done with increasing accuracy as the outer iteration proceeds. In this paper, this idea is extended to generalized problems and is improved by a new two-phase algorithm. Specifically, we show that tuning can be limited to just one step of preconditioned block-GMRES to get an approximate solution, after which a correction equation can be solved to a *fixed* relative tolerance with proper preconditioned block linear solvers where tuning is *not* needed. We show that the effect of tuning is to reduce the residual in a special way, and that this effect can be also achieved by other means, in particular by solving a small least squares problem. Moreover, we show that the two-phase strategy is closely related to an inverse correction scheme presented in [29,17] and the residual inverse power method in [34].

The second phase of this algorithm, in addition to using a simplified preconditioning strategy, can also be simplified in other ways to achieve additional reduction of inner iteration cost. We explore three techniques to attain the extra speedup:

1. *Deflation of converged Schur vectors* (see [33]) – Once some Schur vectors have converged, they are deflated from the block linear systems in subsequent outer iterations, so that the block size becomes smaller. This approach is independent of the way the block linear systems are solved.
2. *Special starting vector* – We find that the right hand sides of a few successive correction equations are often close to being linearly dependent; therefore an appropriate linear combination of the solutions to previously solved correction equations can be used as a good starting vector for solving the current one.

3. *Subspace recycling* – Linear solvers with recycled subspaces (see [27]) can be used to solve the sequence of correction equations, so that the search space for each solve does not need to be built from scratch. In addition, if the same preconditioner is used for all correction equations, the recycled subspaces available from solving one equation can be used directly for the next without being transformed by additional preconditioned matrix–vector products.

We discuss the effectiveness of these ideas and show by numerical experiments that they generally result in significant savings in the number of preconditioned matrix–vector products performed in inner iterations.

An outline of the paper is as follows. In Section 2, we describe the inexact subspace iteration for generalized non-Hermitian eigenvalue problems, restate some preliminary results taken from [28] about block decomposition of matrices, and discuss a new tool for measuring closeness of two subspaces. In Section 3, we briefly discuss the behavior of unpreconditioned and preconditioned block-GMRES without tuning for solving the block linear systems arising in inexact subspace iteration, and present new insights into tuning that lead to our two-phase strategy to solve the block linear systems. In Section 4, we discuss deflation of converged Schur vectors, special starting vector and linear solvers with recycled subspaces and the effectiveness of the combined use of these techniques for solving the block systems. Section 5 includes a series of numerical experiments to show the performance of our algorithm for problems from Matrix Market [22] and those arising from linear stability analysis of models of two-dimensional incompressible flows. We finally draw conclusions in Section 6.

2. Inexact subspace iteration and preliminary results

In this section, we review inexact subspace iteration for the generalized non-Hermitian eigenvalue problem $Av = \lambda Bv$ ($A, B \in \mathbb{C}^{n\times n}$) with spectral transformation, block Schur and eigen-decomposition of matrices, and metrics that measure the error of the current approximate invariant subspace.

2.1. Spectral transformation and inexact subspace iteration

To better understand the algorithm, we start with the definition of the shift-invert and the generalized Cayley transformation (see [23]) as follows:

$$\begin{aligned} Av = \lambda Bv &\Leftrightarrow (A-\sigma B)^{-1}Bv = \frac{1}{\lambda-\sigma}v \quad \text{(shift-invert)} \\ Av = \lambda Bv &\Leftrightarrow (A-\sigma_1 B)^{-1}(A-\sigma_2 B)v = \frac{\lambda-\sigma_2}{\lambda-\sigma_1}v \quad \text{(generalized Cayley)} \end{aligned} \tag{2.1}$$

The shift-invert transformation maps eigenvalues closest to σ to dominant eigenvalues of $\mathcal{A} = (A-\sigma B)^{-1}B$; the generalized Cayley transformation maps eigenvalues to the right of the line $\Re(\lambda) = \frac{\sigma_1+\sigma_2}{2}$ to eigenvalues of $\mathcal{A} = (A-\sigma_1 B)^{-1}(A-\sigma_2 B)$ outside the unit circle, and those to the left of this line to ones inside the unit circle (assuming that $\sigma_1 > \sigma_2$). The eigenvalues of $\mathcal{A}$ with largest magnitude can be easily found by iterative eigenvalue algorithms. Once the eigenvalues of the transformed problem are found, they are transformed back to those of the original problem. Note that the eigenvectors do not change with the transformation.

Without loss of generality, we consider using $\mathcal{A} = A^{-1}B$ for inexact subspace iteration to compute k eigenvalues of $Av = \lambda Bv$ with smallest magnitude (i.e., k dominant eigenvalues of $\mathcal{A} = A^{-1}B$). This notation covers both types of operators in (2.1) with arbitrary shifts. For example, one can let $\widehat{A} = A - \sigma_1 B$ and $\widehat{B} = A - \sigma_2 B$, so that the generalized Cayley operator is $\mathcal{A} = \widehat{A}^{-1}\widehat{B}$. The algorithm is as follows:

Algorithm 1. Inexact subspace iteration with $\mathcal{A} = A^{-1}B$.

```
Given $\delta \geqslant 0$ and $X^{(0)} \in \mathbb{C}^{n\times p}$ with $X^{(0)*}X^{(0)} = I \ (k \leqslant p)$
For $i = 0, 1, \ldots,$ until $k$ Schur vectors converge
    1. Compute the error $e^{(i)} = \sin\angle(A\mathcal{X}^{(i)}, B\mathcal{X}^{(i)})$
    2. Solve $AY^{(i)} = BX^{(i)}$ inexactly such that the relative residual norm $\frac{\|BX^{(i)}-AY^{(i)}\|}{\|BX^{(i)}\|} \leqslant \delta e^{(i)}$
    3. Perform the Schur–Rayleigh–Ritz procedure to get $X^{(i+1)}$ with orthonormal columns
       from $Y^{(i)}$ and test for convergence
End For
```

In Step 1, $\mathcal{X}^{(i)}$ is the space spanned by the current outer iterate $X^{(i)}$. The error of $X^{(i)}$ is defined as the sine of the largest principal angle between $A\mathcal{X}^{(i)}$ and $B\mathcal{X}^{(i)}$. It decreases to zero as $X^{(i)}$ converges to an invariant subspace of the matrix pencil (A, B). This error can be computed by MATLAB's function `subspace` based on singular value decomposition (see Algorithm 12.4.3 of [16]), and will be discussed in detail in Proposition 2.2.

The Schur–Rayleigh–Ritz (SRR) procedure (see Chapter 6.1 of [33]) in Step 3 will be applied to deflate converged Schur vectors. The procedure and its use for deflation will be explained in detail in Section 4. The most computationally expensive part of the algorithm is Step 2, which requires an inexact solve of the block linear system $AY^{(i)} = BX^{(i)}$. The major concern of this paper is to reduce the cost of this solve.

2.2. Block eigen-decomposition

To briefly review the basic notations and description of the generalized eigenvalue problem, we restate some results from [28] on block eigen-decomposition of matrices and study a new tool to measure the error of $X^{(i)}$ for generalized problems. To simplify our exposition, we assume that B is nonsingular. This assumption is valid for problems arising in a variety of applications. In addition, though B is only positive semi-definite in linear stability analysis of incompressible flows, one can instead solve some related eigenvalue problems with nonsingular B that share the same finite eigenvalues as the original problem; see [4] for details.

As B is nonsingular, we assume that the eigenvalues of $B^{-1}A$ are ordered so that

$$0 < |\lambda_1| \leqslant |\lambda_2| \leqslant \ldots \leqslant |\lambda_p| < |\lambda_{p+1}| \leqslant \ldots \leqslant |\lambda_n|.$$

The Schur decomposition of $B^{-1}A$ can be written in block form as

$$B^{-1}A = \left[V_1,\ V_1^{\perp}\right]\begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22}\end{bmatrix}\left[V_1,\ V_1^{\perp}\right]^*, \tag{2.2}$$

where $\left[V_1,\ V_1^{\perp}\right]$ is a unitary matrix with $V_1 \in \mathbb{C}^{n\times p}$ and $V_1^{\perp} \in \mathbb{C}^{n\times(n-p)}$, $T_{11} \in \mathbb{C}^{p\times p}$ and $T_{22} \in \mathbb{C}^{(n-p)\times(n-p)}$ are upper triangular, $\lambda(T_{11}) = \{\lambda_1, \ldots, \lambda_p\}$ and $\lambda(T_{22}) = \{\lambda_{p+1}, \ldots, \lambda_n\}$. Since T_{11} and T_{22} have disjoint spectra, there is a unique solution $Q \in \mathbb{C}^{p\times(n-p)}$ to the Sylvester equation $QT_{22} - T_{11}Q = T_{12}$ (see Section 1.5, Chapter 1 of [33]). Then $B^{-1}A$ can be transformed to block-diagonal form as follows:

$$\begin{aligned} B^{-1}A &= \left[V_1,\ V_1^{\perp}\right]\begin{bmatrix} I & Q \\ 0 & I\end{bmatrix}\begin{bmatrix} T_{11} & 0 \\ 0 & T_{22}\end{bmatrix}\begin{bmatrix} I & -Q \\ 0 & I\end{bmatrix}\left[V_1,\ V_1^{\perp}\right]^* \\ &= \left[V_1,\ (V_1Q + V_1^{\perp})\right]\begin{bmatrix} T_{11} & 0 \\ 0 & T_{22}\end{bmatrix}\left[(V_1 - V_1^{\perp}Q^*),\ V_1^{\perp}\right]^* \\ &= \left[V_1,\ (V_1Q + V_1^{\perp})Q_D^{-1}\right]\begin{bmatrix} T_{11} & 0 \\ 0 & Q_DT_{22}Q_D^{-1}\end{bmatrix}\left[(V_1 - V_1^{\perp}Q^*),\ V_1^{\perp}Q_D\right]^* \\ &= [V_1,\ V_2]\begin{bmatrix} K & 0 \\ 0 & M\end{bmatrix}[W_1,\ W_2]^* = [V_1,\ V_2]\begin{bmatrix} K & 0 \\ 0 & M\end{bmatrix}[V_1,\ V_2]^{-1}, \end{aligned} \tag{2.3}$$

where $Q_D = (I + Q^*Q)^{1/2}$, $V_2 = (V_1 Q + V_1^{\perp})Q_D^{-1}$ with orthonormal columns, $K = T_{11}$, $M = Q_D T_{22} Q_D^{-1}$ with the same spectrum as T_{22}, $W_1 = V_1 - V_1^{\perp} Q^*$ and $W_2 = V_1^{\perp} Q_D$ such that $[W_1,\ W_2]^* = [V_1,\ V_2]^{-1}$. From the last expression of (2.3), we have

$$AV_1 = BV_1 K, \quad \text{and} \quad AV_2 = BV_2 M. \tag{2.4}$$

Recall that we want to compute V_1 and corresponding eigenvalues (the spectrum of K) by inexact subspace iteration.

2.3. Tools to measure the error

The basic tool to measure the deviation of $\mathcal{X}^{(i)} = \text{span}\{X^{(i)}\}$ from $\mathcal{V}_1 = \text{span}\{V_1\}$ is the sine of the largest principal angle between $\mathcal{X}^{(i)}$ and $\mathcal{V}_1$ defined as (see [28] and references therein)

$$\begin{aligned}\sin(\mathcal{X}^{(i)}, \mathcal{V}_1) &= \|(V_1^{\perp})^* X^{(i)}\| = \|X^{(i)}(X^{(i)})^* - V_1 V_1^*\| \\ &= \min_{Z \in \mathbb{C}^{p \times p}} \|X^{(i)} - V_1 Z\| = \min_{Z \in \mathbb{C}^{p \times p}} \|V_1 - X^{(i)} Z\|.\end{aligned} \tag{2.5}$$

This definition depends on the fact that both $X^{(i)}$ and V_1 have orthonormal columns.

We assume that $X^{(i)}$ has the following decomposition

$$X^{(i)} = V_1 C^{(i)} + V_2 S^{(i)} \quad \text{with } \|S^{(i)}\| < 1, \tag{2.6}$$

where $C^{(i)} = W_1^* X^{(i)} \in \mathbb{C}^{p \times p}$, $S^{(i)} = W_2^* X^{(i)} \in \mathbb{C}^{(n-p) \times p}$. Intuitively, $\|S^{(i)}\| \to 0$ as $\mathcal{X}^{(i)} \to \mathcal{V}_1$. Properties of $C^{(i)}$ and the equivalence of several metrics are given in the following proposition.

Proposition 2.1 (Proposition 2.1 in [28]). *Suppose $X^{(i)}$ is decomposed as in (2.6). Let $s^{(i)} = \|S^{(i)}(C^{(i)})^{-1}\|$ and $t^{(i)} = s^{(i)}\|C^{(i)}\|$. Then*

(1) *$C^{(i)}$ is nonsingular and thus $t^{(i)}$ is well-defined. The singular values of $C^{(i)}$ satisfy*

$$0 < 1 - \|S^{(i)}\| \leqslant \sigma_k(C^{(i)}) \leqslant 1 + \|S^{(i)}\|, \ k = 1, 2, \ldots, p \tag{2.7}$$

and $C^{(i)} = U^{(i)} + \Upsilon^{(i)}$, where $U^{(i)}$ is unitary and $\|\Upsilon^{(i)}\| \leqslant \|S^{(i)}\| < 1$.

(2a) $\sin(\mathcal{X}^{(i)}, \mathcal{V}_1) \leqslant \|S^{(i)}\| \leqslant s^{(i)} \leqslant \left(\frac{1+\|S^{(i)}\|}{1-\|S^{(i)}\|}\right)\|S^{(i)}\|$.

(2b) $\sin(\mathcal{X}^{(i)}, \mathcal{V}_1) \leqslant t^{(i)} \leqslant \frac{\|S^{(i)}\|}{1-\|S^{(i)}\|}$.

(2c) $\|S^{(i)}\| \leqslant \sqrt{1 + \|Q\|^2}\,\sin(\mathcal{X}^{(i)}, \mathcal{V}_1)$.

The proposition states that as $\mathcal{X}^{(i)} \to \mathcal{V}_1$, $C^{(i)}$ gradually approximates a unitary matrix, and $\sin(\mathcal{X}^{(i)}, \mathcal{V}_1)$, $\|S^{(i)}\|$, $s^{(i)}$ and $t^{(i)}$ are essentially equivalent measures of the error. These quantities are not computable since $\mathcal{V}_1$ is not available. However, the computable quantity $\sin(A\mathcal{X}^{(i)}, B\mathcal{X}^{(i)})$ in Step 1 of Algorithm 1 is equivalent to $\|S^{(i)}\|$, as the following proposition shows.

Proposition 2.2. *Let $X^{(i)}$ be decomposed as in (2.6). Then*

$$c_1\|S^{(i)}\| \leqslant \sin(A\mathcal{X}^{(i)}, B\mathcal{X}^{(i)}) \leqslant c_2\|S^{(i)}\|, \tag{2.8}$$

where c_1 and c_2 are constants independent of the progress of subspace iteration.

Proof. We first show that as $\mathcal{X}^{(i)} \to \mathcal{V}_1, A\mathcal{X}^{(i)} \approx B\mathcal{X}^{(i)}$. In fact, from (2.4) and (2.6) we have

$$\begin{aligned} BX^{(i)} &= BV_1C^{(i)} + BV_2S^{(i)} = AV_1K^{-1}C^{(i)} + AV_2M^{-1}S^{(i)} \\ &= A(X^{(i)} - V_2S^{(i)})(C^{(i)})^{-1}K^{-1}C^{(i)} + AV_2M^{-1}S^{(i)} \\ &= AX^{(i)}(C^{(i)})^{-1}K^{-1}C^{(i)} - AV_2\left(S^{(i)}(C^{(i)})^{-1}K^{-1}C^{(i)} - M^{-1}S^{(i)}\right). \end{aligned} \tag{2.9}$$

Roughly speaking, $AX^{(i)}$ and $BX^{(i)}$ can be transformed to each other by postmultiplying $(C^{(i)})^{-1}K^{-1}C^{(i)}$ or its inverse, with a small error proportional to $\|S^{(i)}\|$.

Let $D_A^{(i)} = (X^{(i)*}A^*AX^{(i)})^{-1/2}$, $D_B^{(i)} = (X^{(i)*}B^*BX^{(i)})^{-1/2} \in \mathbb{C}^{p\times p}$, so that both $AX^{(i)}D_A^{(i)}$ and $BX^{(i)}D_B^{(i)}$ have orthonormal columns. Then by (2.5)

$$\begin{aligned} \sin\angle(A\mathcal{X}^{(i)}, B\mathcal{X}^{(i)}) &= \min_{Z\in\mathbb{C}^{p\times p}} \|AX^{(i)}D_A^{(i)} - BX^{(i)}D_B^{(i)}Z\| \\ &\leqslant \left\|AX^{(i)}D_A^{(i)} - BX^{(i)}D_B^{(i)}\left((D_B^{(i)})^{-1}(C^{(i)})^{-1}KC^{(i)}D_A^{(i)}\right)\right\| \\ &= \left\|\left(AX^{(i)} - BX^{(i)}(C^{(i)})^{-1}KC^{(i)}\right)D_A^{(i)}\right\| \\ &= \left\|BV_2\left(MS^{(i)} - S^{(i)}(C^{(i)})^{-1}KC^{(i)}\right)D_A^{(i)}\right\| \qquad \text{(see (2.9) and (2.4))} \\ &\leqslant \|BV_2\|\|D_A^{(i)}\|\|\mathcal{S}_i\|\|S^{(i)}\|, \end{aligned} \tag{2.10}$$

where $\mathcal{S}_i$ is the Sylvester operator $G \to \mathcal{S}_i(G) : MG - G(C^{(i)})^{-1}KC^{(i)}$. Note that as i increases, $\|\mathcal{S}_i\| \to \|\mathcal{S}\|$ where $\mathcal{S} : G \to \mathcal{S}(G) = MG - GK$. In fact,

$$\begin{aligned} \|\mathcal{S}_i\| &= \sup_G \frac{\|MG - G(C^{(i)})^{-1}KC^{(i)}\|}{\|G\|} \qquad (G \in \mathbb{C}^{(n-p)\times p}) \\ &= \sup_G \frac{\left\|\left(M\left(G(C^{(i)})^{-1}\right) - \left(G(C^{(i)})^{-1}\right)K\right)C^{(i)}\right\|}{\|\left(G(C^{(i)})^{-1}\right)C^{(i)}\|}, \end{aligned} \tag{2.11}$$

and therefore

$$\begin{aligned} &\sup_{\widetilde{G}} \frac{\|M\widetilde{G} - \widetilde{G}K\|}{\|\widetilde{G}\|\kappa(C^{(i)})} \leqslant \|\mathcal{S}_i\| \leqslant \sup_{\widetilde{G}} \frac{\|M\widetilde{G} - \widetilde{G}K\|\kappa(C^{(i)})}{\|\widetilde{G}\|} \quad (\widetilde{G} \in \mathbb{C}^{(n-p)\times p}) \\ \text{or} \quad &\|S\|/\kappa(C^{(i)}) \leqslant \|\mathcal{S}_i\| \leqslant \|S\|\kappa(C^{(i)}). \end{aligned} \tag{2.12}$$

As $1 \leqslant \kappa(C^{(i)}) \leqslant \frac{1+\|S^{(i)}\|}{1-\|S^{(i)}\|}$ (Proposition 2.1, 2a) and $\|S^{(i)}\| \to 0$, $\|\mathcal{S}_i\| \to \|\mathcal{S}\|$ follows. Also, note that the extremal singular values of $D_A^{(i)}$ are bounded by those of $(A^*A)^{-1/2}$ (equivalently, those of A^{-1}), so that in particular $\|D_A^{(i)}\|$ is bounded above by a constant independent of i. The upper bound in (2.8) is thus established.

To study the lower bound, we have

$$\begin{aligned} \sin\angle(A\mathcal{X}^{(i)}, B\mathcal{X}^{(i)}) &= \min_{Z\in\mathbb{C}^{p\times p}} \|AX^{(i)}D_A^{(i)} - BX^{(i)}D_B^{(i)}Z\| \\ &= \min_{Z\in\mathbb{C}^{p\times p}} \left\|B\left(B^{-1}AX^{(i)} - X^{(i)}D_B^{(i)}Z(D_A^{(i)})^{-1}\right)D_A^{(i)}\right\| \\ &\geqslant \min_{Z\in\mathbb{C}^{p\times p}} \|B^{-1}AX^{(i)} - X^{(i)}Z\|\sigma_{min}(B)\sigma_{min}(D_A^{(i)}). \end{aligned} \tag{2.13}$$

Let $\sigma^{(i)} = \sigma_{min}(B)\sigma_{min}(D_A^{(i)}) = \sigma_{min}(B)\|X^{(i)*}A^*AX^{(i)}\|^{-1/2}$. Again using the boundedness of the singular values of $D_A^{(i)}$, it follows that $\sigma^{(i)}$ is bounded below by a constant independent of i. Since the minimizer Z in the inequality (2.13) is $K^{(i)} = X^{(i)*}B^{-1}AX^{(i)}$ (see Chapter 4, Theorem 2.6 of [33]), we have

$$\begin{aligned}
\sin\angle(A\mathcal{X}^{(i)}, B\mathcal{X}^{(i)}) &\geqslant \sigma^{(i)}\|B^{-1}AX^{(i)} - X^{(i)}K^{(i)}\| \\
&\geqslant \sigma^{(i)}\|(V_1^{\perp})^*(B^{-1}AX^{(i)} - X^{(i)}K^{(i)})\| \quad (\,\|V_1^{\perp}\| = 1\,) \\
&= \sigma^{(i)}\|T_{22}(V_1^{\perp})^*X^{(i)} - (V_1^{\perp})^*X^{(i)}K^{(i)}\| \quad (\,(V_1^{\perp})^*B^{-1}A = T_{22}(V_1^{\perp})^*;\ \text{see (2.2)}) \\
&\geqslant \sigma^{(i)}\mathrm{sep}(T_{22}, K^{(i)})\|(V_1^{\perp})^*X^{(i)}\| = \sigma^{(i)}\mathrm{sep}(T_{22}, K^{(i)})\sin\angle(\mathcal{V}_1, \mathcal{X}^{(i)}) \\
&\geqslant \sigma^{(i)}\mathrm{sep}(T_{22}, K^{(i)})(1 + \|Q\|^2)^{-1/2}\|S^{(i)}\|. \quad \text{(Proposition 2.1(2c))} \qquad (2.14)
\end{aligned}$$

Moreover, as $\mathcal{X}^{(i)} \to \mathcal{V}_1$, $K^{(i)} = X^{(i)*}B^{-1}AX^{(i)} \to K$ up to a unitary transformation, and hence $\mathrm{sep}(T_{22}, K^{(i)}) \to \mathrm{sep}(T_{22}, K)$ (see Chapter 4, Theorem 2.11 in [33]). This concludes the proof. □

Remark 2.1. In [28] the authors use $\|AX^{(i)} - X^{(i)}(X^{(i)*}AX^{(i)})\|$ to estimate the error of $X^{(i)}$ for standard eigenvalue problems, and show that this quantity is essentially equivalent to $\|S^{(i)}\|$. The $p \times p$ matrix $X^{(i)*}AX^{(i)}$ is called the (block) Rayleigh quotient of A (see Chapter 4, Definition 2.7 of [33]). For the generalized eigenvalue problem, we have not seen an analogous quantity in the literature. However, Proposition 2.2 shows that $\sin\angle(A\mathcal{X}^{(i)}, B\mathcal{X}^{(i)})$ is a convenient error estimate.

3. Convergence analysis of inexact subspace iteration

We first demonstrate the linear convergence of the outer iteration of Algorithm 1, which follows from Theorem 3.1 in [28]. In fact, replacing A^{-1} by $A^{-1}B$ in the proof therein, we have

$$t^{(i+1)} \leqslant \|K\|\|M^{-1}\|\frac{t^{(i)} + \|W_2^*B^{-1}\|\,\|(C^{(i)})^{-1}\|\,\|R^{(i)}\|}{1 - \|W_1^*B^{-1}\|\,\|(C^{(i)})^{-1}\|\,\|R^{(i)}\|}, \qquad (3.1)$$

where $\|(C^{(i)})^{-1}\| = 1/\sigma_{min}(C^{(i)}) \to 1$, and

$$\begin{aligned}
\|R^{(i)}\| &= \|BX^{(i)} - AY^{(i)}\| \leqslant \delta\|BX^{(i)}\|\sin\angle(A\mathcal{X}^{(i)}, B\mathcal{X}^{(i)}) \\
&\leqslant \delta C_2\|BX^{(i)}\|\sqrt{1 + \|Q\|^2}\,t^{(i)}. \quad \text{(see Proposition 2.1)} \qquad (3.2)
\end{aligned}$$

Thus $\mathcal{X}^{(i)} \to \mathcal{V}_1$ linearly for $\|K\|\|M^{-1}\| < 1$ and small enough δ.

In this section, we investigate the convergence of unpreconditioned and preconditioned block-GMRES without tuning for solving $AY^{(i)} = BX^{(i)}$ to the prescribed tolerance, and provide new perspectives on tuning that lead to a new two-phase strategy for solving this block system.

3.1. Unpreconditioned and preconditioned block-GMRES with no tuning

The block linear systems $AY^{(i)} = X^{(i)}$ arising in inexact subspace iteration for standard eigenvalue problems are solved to increasing accuracy as the outer iteration progresses. It is shown in [28] that when unpreconditioned block-GMRES is used for these systems, iteration counts remain roughly constant during the course of this computation, but for preconditioned block-GMRES without tuning, the number of iterations increases with the outer iteration. The reason is that the right hand side $X^{(i)}$ is an approximate invariant subspace of the system matrix A, whereas with preconditioning, there is no reason for $X^{(i)}$ to bear such a relation to the preconditioned system matrix AP^{-1} (assuming right preconditioning is used).

For generalized eigenvalue problems, however, both the unpreconditioned and preconditioned block-GMRES iteration counts increase progressively. To see this point, we study the block spectral decomposition of the system matrix and right hand side and review the block-GMRES convergence theory. We present the analysis of the unpreconditioned solve; the results apply verbatim to the preconditioned solve.

We first review a generic convergence result of block-GMRES given in [28]. Let G be a matrix of order n where the p smallest eigenvalues are separated from its other $n-p$ eigenvalues. As in (2.3), we can block diagonalize G as

$$G = [V_{G1},\ V_{G2}] \begin{bmatrix} K_G & 0 \\ 0 & M_G \end{bmatrix} [V_{G1},\ V_{G2}]^{-1}, \tag{3.3}$$

where $V_{G1} \in \mathbb{C}^{n\times p}$ and $V_{G2} \in \mathbb{C}^{n\times(n-p)}$ have orthonormal columns, $\lambda(K_G)$ are the p smallest eigenvalues of G, and $\lambda(M_G)$ are the other eigenvalues of G. Recall the definitions of the numerical range $W(M_G) = \left\{\frac{z^*M_Gz}{z^*z} : z \in \mathbb{C}^{n-p}, z \neq 0\right\}$ and the ϵ-pseudospectrum $\lambda_\epsilon(M_G) = \{\lambda \in \mathbb{C} : \sigma_{min}(\lambda I - M_G) \leqslant \epsilon\}$. The role of the right hand side in the convergence of block-GMRES is described in the following lemma, which follows immediately from Theorem 3.7 of [28].

Lemma 3.1. *Assume the numerical range $W(M_G)$ or the ϵ-pseudospectrum $\lambda_\epsilon(M_G)$ is contained in a convex closed bounded set E in the complex plane with $0 \notin E$. Suppose block-GMRES is used to solve $GY = Z$ where $Z \in \mathbb{C}^{n\times p}$ can be decomposed as $Z = V_{G1}C_G + V_{G2}S_G$ with V_{G1} and V_{G2} given in (3.3). Here $S_G \in \mathbb{C}^{(n-p)\times p}$, and we assume $C_G \in \mathbb{C}^{p\times p}$ is nonsingular. Let Y_k be the approximate solution of $GY = Z$ obtained in the k-th block-GMRES iteration with $Y_0 = 0$. If*

$$k \geqslant 1 + C_a\left(C_b + \log\frac{\|C_G\|\|S_GC_G^{-1}\|}{\|Z\|\tau}\right), \tag{3.4}$$

then $\frac{\|Z-GY_k\|}{\|Z\|} \leqslant \tau$. Here C_a and C_b are constants that depend on the spectrum of G.

Remark 3.1. For details about C_a and C_b, see [28,18,11]. These details have minimal impact on our subsequent analysis.

Remark 3.2. This generic convergence result can be applied to any specific block linear systems with or without preconditioning. For example, to study the behavior of unpreconditioned block-GMRES for solving $AY^{(i)} = BX^{(i)}$, let $G = A$ in (3.3) and Lemma 3.1, and decompose relevant sets of column vectors in terms of V_{A1} and V_{A2}.

In the following, we will assume that for nontrivial B ($B \neq I$), there is no pathological or trivial connection between the decomposition of (3.3) for the case $G = A$ and that of $B^{-1}A$ in (2.3). Specifically, we assume there exist a *nonsingular* $C_1 \in \mathbb{C}^{p\times p}$ and a *full rank* $S_1 \in \mathbb{C}^{(n-p)\times p}$ such that

$$V_1 = V_{A1}C_1 + V_{A2}S_1. \tag{3.5}$$

This assumption is generally far from stringent in practice. Similarly, let the decomposition of V_2 be $V_2 = V_{A1}C_2 + V_{A2}S_2$ with $C_2 \in \mathbb{C}^{p\times(n-p)}$ and $S_2 \in \mathbb{C}^{(n-p)\times(n-p)}$.

Lemma 3.1 and the assumptions above lead to the following theorem, which gives qualitative insight into the behavior of block-GMRES for solving $AY^{(i)} = BX^{(i)}$.

Theorem 3.2. *Assume that unpreconditioned block-GMRES is used to solve the linear system $AY^{(i)} = BX^{(i)}$ to the prescribed tolerance in Step 2 of Algorithm 1. If the assumption (3.5) holds, and $\mathcal{X}^{(i)} \to \mathcal{V}_1$ linearly, then the lower bound on block-GMRES iterations from Lemma 3.1 increases as the outer iteration proceeds.*

Proof. With (2.4), (2.6), (3.3) and (3.5), we have

$$\begin{aligned} BX^{(i)} &= BV_1C^{(i)} + BV_2S^{(i)} = AV_1K^{-1}C^{(i)} + AV_2M^{-1}S^{(i)} \\ &= A(V_{A1}C_1 + V_{A2}S_1)K^{-1}C^{(i)} + A(V_{A1}C_2 + V_{A2}S_2)M^{-1}S^{(i)} \\ &= (V_{A1}K_AC_1 + V_{A2}M_AS_1)K^{-1}C^{(i)} + (V_{A1}K_AC_2 + V_{A2}M_AS_2)M^{-1}S^{(i)} \\ &= V_{A1}K_A(C_1K^{-1}C^{(i)} + C_2M^{-1}S^{(i)}) + V_{A2}M_A(S_1K^{-1}C^{(i)} + S_2M^{-1}S^{(i)}) \\ &= V_{A1}C_A^{(i)} + V_{A2}S_A^{(i)}. \end{aligned} \tag{3.6}$$

Since $\|S^{(i)}\| \to 0$ and $\sigma_k(C^{(i)}) \to 1 (k = 1, 2, \ldots, p)$ (see Proposition 2.1(1)), we have $\|C_A^{(i)}\| \to \|K_AC_1K^{-1}\|$ and $\|S_A^{(i)}(C_A^{(i)})^{-1}\| \to \|M_AS_1C_1^{-1}K_A^{-1}\|$, both of which are nonzero under our assumption (3.5).

From Step 2 of Algorithm 1 and (2.10), the relative tolerance for $AY^{(i)} = BX^{(i)}$ is $\tau = \delta \sin(A\mathcal{X}^{(i)}, B\mathcal{X}^{(i)}) \leqslant \delta\|BV_2\|\|D_A^{(i)}\|\|\mathcal{S}_i\|\|S^{(i)}\| \to 0$. Then from (3.4) and (3.6), the lower bound of the unpreconditioned block-GMRES iteration counts needed for solving $AY^{(i)} = BX^{(i)}$ to the prescribed tolerance is

$$k^{(i)} \geqslant 1 + C_a\left(C_b + \log\frac{\|C_A^{(i)}\|\|S_A^{(i)}(C_A^{(i)})^{-1}\|}{\delta\|BX^{(i)}\|\|BV_2\|\|D_A^{(i)}\|\|\mathcal{S}_i\|\|S^{(i)}\|}\right). \tag{3.7}$$

Note that $\|BX^{(i)}\| \to \|BV_1\|$, $\|D_A^{(i)}\| \to \|(V_1A^*AV_1)^{-1/2}\|$ and $\|\mathcal{S}_i\| \to \|\mathcal{S}\|$ (see the proof of Proposition 2.2). Therefore all terms in the argument of the logarithm operator approach some nonzero

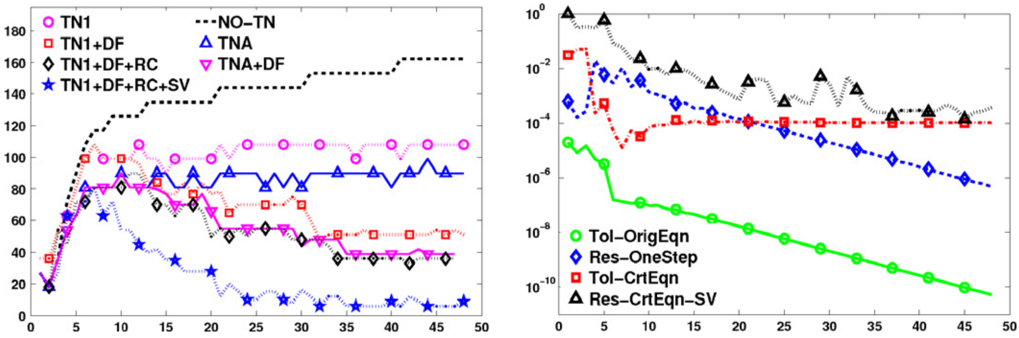

Fig. 1. Performance of different solution strategies for Problem 1. (a) Preconditioned matrix–vector product counts of the inner iteration against the outer iteration. (b) Behavior of the two-phase strategy and starting vector.

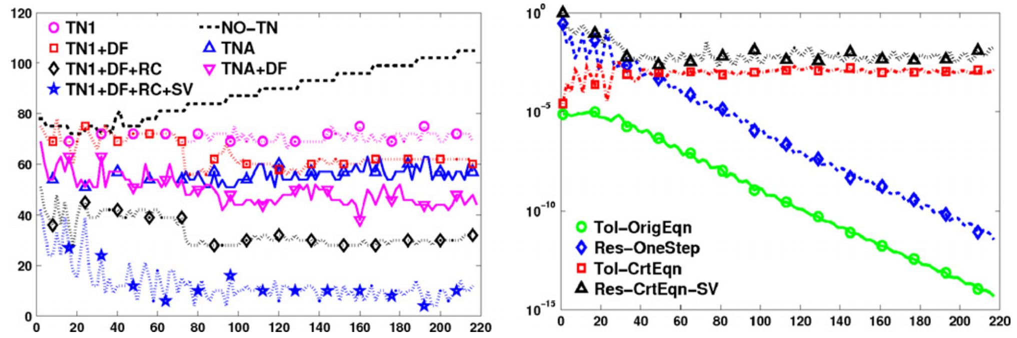

Fig. 2. Performance of different solution strategies for Problem 2. (a) Preconditioned matrix–vector product counts of the inner iteration against the outer iteration. (b) Behavior of the two-phase strategy and starting vector.

limit except $\|S^{(i)}\| \to 0$, and hence the expression on the right in (3.7) increases as the outer iteration proceeds. $\square$

Remark 3.3. This result only shows that a lower bound on $k^{(i)}$ increases; it does not establish that there will be growth in actual iteration counts. However, numerical experiments described in [28] and Section 5 (see Figs. 1 and 2) show that this result is indicative of performance. The fact that the bound progressively increases depends on the assumption of (3.5) that V_1 has "regular" components of V_{A1} and V_{A2}. This assumption guarantees that $B\mathcal{X}^{(i)}$ does not approximate the invariant subspace $\mathcal{V}_{A1}$ of A. The proof also applies word for word to the preconditioned solve without tuning: one only needs to replace (3.3) by the decomposition of the preconditioned system matrix AP^{-1} and write $BX^{(i)}$ in terms of the invariant subspace of AP^{-1}.

3.2. Preconditioned block-GMRES with tuning

To accelerate the iterative solution of the block linear system arising in inexact subspace iteration, [28] proposes and analyzes a new type of preconditioner with tuning. Tuning constructs a special low-rank update of the existing preconditioner, so that the right hand side of the preconditioned system is an approximate eigenvector or invariant subspace of the preconditioned system matrix with tuning. Specifically, the tuned preconditioner is

$$\mathbb{P}^{(i)} = P + (AX^{(i)} - PX^{(i)})X^{(i)*}, \tag{3.8}$$

from which follow $\mathbb{P}^{(i)}X^{(i)} = AX^{(i)}$ and $A(\mathbb{P}^{(i)})^{-1}(AX^{(i)}) = AX^{(i)}$. In other words, $A\mathcal{X}^{(i)}$ is an invariant subspace of $A(\mathbb{P}^{(i)})^{-1}$ with eigenvalue 1. Intuitively, as $\mathcal{X}^{(i)} \to \mathcal{V}_1$, we have $\mathcal{X}^{(i)} \approx A\mathcal{X}^{(i)}$ for the standard eigenvalue problem, or $B\mathcal{X}^{(i)} \approx A\mathcal{X}^{(i)}$ for the generalized problem. Therefore, the right hand side of $A(\mathbb{P}^{(i)})^{-1}\widetilde{Y}^{(i)} = X^{(i)}$ or $A(\mathbb{P}^{(i)})^{-1}\widetilde{Y}^{(i)} = BX^{(i)}$ (with $Y^{(i)} = (\mathbb{P}^{(i)})^{-1}\widetilde{Y}^{(i)}$) spans an approximate invariant subspace of $A(\mathbb{P}^{(i)})^{-1}$. The difficulty of block-GMRES without tuning discussed in Section 3.1 is thus resolved, and the block-GMRES iteration counts with tuning do not increase with the progress of the outer iteration (see Theorem 4.5 of [28]).

The matrix–vector product involving $(\mathbb{P}^{(i)})^{-1}$ is built from that for P^{-1} using the Sherman–Morrison–Woodbury formula as follows:

$$(\mathbb{P}^{(i)})^{-1} = \left(I - (P^{-1}AX^{(i)} - X^{(i)})\left(X^{(i)*}P^{-1}AX^{(i)}\right)^{-1} X^{(i)*}\right) P^{-1}. \tag{3.9}$$

Note that $P^{-1}AX^{(i)} - X^{(i)}$ and $X^{(i)*}P^{-1}AX^{(i)}$ can be computed before the block-GMRES iteration. In each block-GMRES step, $(\mathbb{P}^{(i)})^{-1}$ requires an additional p^2 inner products of vectors of length n, a dense matrix–matrix division of size $p \times p$, and a multiplication of a dense matrix of size $n \times p$ with a $p \times p$ matrix. This extra cost is relatively small in general, but it is not free.

We now provide a new two-phase algorithm for solving $AY^{(i)} = BX^{(i)}$, which essentially eliminates the overhead of tuning but keeps the block-GMRES iteration counts from progressively increasing. The strategy provides some new perspectives on the use of tuning. In addition, we will discuss some connections between this algorithm and the methods in [29,17,34].

Algorithm 2. Two-phase strategy for solving $AY^{(i)} = BX^{(i)}$.

1. Apply a *single* step of preconditioned block-GMRES with tuning to get an approximate solution $Y_1^{(i)}$.
2. Solve the *correction equation* $A\,dY^{(i)} = BX^{(i)} - AY_1^{(i)}$ with proper preconditioned iterative solver to get an approximate solution $dY_k^{(i)}$, so that $Y_{k+1}^{(i)} = Y_1^{(i)} + dY_k^{(i)}$ satisfies $\frac{\|BX^{(i)} - AY_{k+1}^{(i)}\|}{\|BX^{(i)}\|} \leqslant \delta \sin(A\mathcal{X}^{(i)}, B\mathcal{X}^{(i)})$.

Note in particular that tuning need *not* be used to solve the correction equation, and thus we can work with a fixed preconditioned system matrix for the correction equation in all outer iterations.

Obviously, Phase II can be equivalently stated as follows: solve $AY^{(i)} = BX^{(i)}$ with proper preconditioned iterative solver and starting vector $Y_1^{(i)}$ from Phase I. The phrasing in Algorithm 2 is intended to illuminate the connection between this strategy and the methods in [29,17,34].

The analysis of Algorithm 2 is given in the following main theorem. For this, we make some assumptions concerning the right hand side of the correction equation analogous to the assumption made for (3.5): with

$$BX^{(i)} - AY_1^{(i)} = V_{A1}C_{ceq}^{(i)} + V_{A2}S_{ceq}^{(i)}, \tag{3.10}$$

where V_{A1} and V_{A2} have orthonormal columns, we assume that $\|S_{ceq}^{(i)}(C_{ceq}^{-1})^{(i)}\| = O(1)$, and that $\|C_{ceq}\|$ is proportional to $\|BX^{(i)} - AY_1^{(i)}\|$. This implies that the term $\frac{\|C_{ceq}^{(i)}\| \, \|S_{ceq}^{(i)}(C_{ceq}^{-1})^{(i)}\|}{\|BX^{(i)}-AY_1^{(i)}\|}$, which appears in (3.4), does not depend on i. We have no proof of these assumptions, but they are consistent with all our numerical experience. In the subsequent derivation, let $e_j \in \mathbb{R}^p$ be a standard unit basis vector with 1 in entry j and zero in other entries.

Theorem 3.3. *Suppose the two-phase strategy (Algorithm 2) is used to solve* $AY^{(i)} = BX^{(i)}$. *Then* $Y_1^{(i)} = X^{(i)}(C^{(i)})^{-1}K^{-1}C^{(i)}F + \Delta^{(i)}$ *where* $F = [f_1, \ldots, f_p]$ *with* $f_j = \operatorname{argmin}_{f \in \mathbb{C}^p}\|BX^{(i)}e_j - A(\mathbb{P}^{(i)})^{-1}BX^{(i)}f\|$ *and* $\|\Delta^{(i)}\| = O(\|S^{(i)}\|)$. *In addition, the residual norm* $\|BX^{(i)} - AY_1^{(i)}\| = O(\|S^{(i)}\|)$. *Thus, if block-GMRES is used to solve the correction equation, the inner iteration counts will not increase with the progress of the outer iteration.*

Proof. The approximate solution to $A(\mathbb{P}^{(i)})^{-1}\widetilde{Y} = BX^{(i)}$ in the k-th block-GMRES iteration starting with zero starting vector is

$$\widetilde{Y}_k^{(i)} \in \operatorname{span}\{BX^{(i)}, A(\mathbb{P}^{(i)})^{-1}BX^{(i)}, \ldots, \left(A(\mathbb{P}^{(i)})^{-1}\right)^{k-1} BX^{(i)}\}. \tag{3.11}$$

It follows from (2.9) and $(\mathbb{P}^{(i)})^{-1}AX^{(i)} = X^{(i)}$ that

$$\begin{aligned} Y_1^{(i)} &= (\mathbb{P}^{(i)})^{-1}\widetilde{Y}_1^{(i)} = (\mathbb{P}^{(i)})^{-1}BX^{(i)}F \\ &= (\mathbb{P}^{(i)})^{-1}\left(AX^{(i)}(C^{(i)})^{-1}K^{-1}C^{(i)} - AV_2\left(S^{(i)}(C^{(i)})^{-1}K^{-1}C^{(i)} - M^{-1}S^{(i)}\right)\right)F \\ &= X^{(i)}(C^{(i)})^{-1}K^{-1}C^{(i)}F + (\mathbb{P}^{(i)})^{-1}AV_2\left(M^{-1}S^{(i)} - S^{(i)}(C^{(i)})^{-1}K^{-1}C^{(i)}\right)F \\ &= X^{(i)}(C^{(i)})^{-1}K^{-1}C^{(i)}F + \Delta^{(i)}, \end{aligned} \tag{3.12}$$

where the jth column of $F \in \mathbb{C}^{p\times p}$ minimizes $\|BX^{(i)}e_j - A(\mathbb{P}^{(i)})^{-1}BX^{(i)}f\|$, i.e., the residual norm of the jth individual system (property of block-GMRES), and

$$\|\Delta^{(i)}\| \leqslant \|(\mathbb{P}^{(i)})^{-1}AV_2\|\,\|F\|\,\|\overline{\mathcal{S}}_i\|\,\|S^{(i)}\|, \tag{3.13}$$

where the Sylvester operator $\overline{\mathcal{S}}_i : G \to \overline{\mathcal{S}}_i(G) = M^{-1}G - G(C^{(i)})^{-1}K^{-1}C^{(i)}$. Using the same derivation as in the proof of Proposition 2.2, we can show $\|\overline{\mathcal{S}}_i\| \to \|\overline{\mathcal{S}}\|$, where $\overline{\mathcal{S}} : G \to \overline{\mathcal{S}}(G) = M^{-1}G - GK^{-1}$. In addition, since $\mathcal{X}^{(i)} \to \mathcal{V}_1$ in (3.8), it follows that $\mathbb{P}^{(i)} \to \mathbb{P} \equiv P + (AV_1 - PV_1)V_1^*$. Thus $\|\Delta^{(i)}\| = O(\|S^{(i)}\|)$ is established.

We now investigate the residual norm $\|BX^{(i)} - AY_1^{(i)}\|$ of the linear system after Phase I of Algorithm 2. Recall the property of tuning that $\left(I - A(\mathbb{P}^{(i)})^{-1}\right)AX^{(i)} = 0$, and the property of block-GMRES that the approximate solution $\widetilde{Y}_1^{(i)} \in \operatorname{span}\{BX^{(i)}\}$. As block-GMRES minimizes the residual norm of each individual linear system of the block system, the j-th column of the block residual is

$$\begin{aligned}\left\|\left(BX^{(i)} - A(\mathbb{P}^{(i)})^{-1}\widetilde{Y}_1^{(i)}\right)e_j\right\| &= \min_{f\in\mathbb{C}^p} \|BX^{(i)}e_j - A(\mathbb{P}^{(i)})^{-1}(BX^{(i)}f)\| \\ &\leqslant \left\|\left(I - A(\mathbb{P}^{(i)})^{-1}\right)BX^{(i)}e_j\right\| \leqslant \left\|\left(I - A(\mathbb{P}^{(i)})^{-1}\right)BX^{(i)}\right\| \\ &= \left\|\left(I - A(\mathbb{P}^{(i)})^{-1}\right)AV_2\left(S^{(i)}(C^{(i)})^{-1}K^{-1}C^{(i)} - M^{-1}S^{(i)}\right)\right\| \qquad (\text{see } (2.9)) \\ &\leqslant \left\|\left(I - A(\mathbb{P}^{(i)})^{-1}\right)AV_2\right\| \|\overline{\mathcal{S}}_i\|\|S^{(i)}\| = O(\|S^{(i)}\|), \end{aligned} \tag{3.14}$$

from which follows

$$\|BX^{(i)} - AY_1^{(i)}\| \leqslant \sum_{j=1}^{p} \left\|\left(BX^{(i)} - A(\mathbb{P}^{(i)})^{-1}\widetilde{Y}_1^{(i)}\right)e_j\right\| = O(\|S^{(i)}\|). \tag{3.15}$$

Finally in Phase II of Algorithm 2, $Y_{k+1}^{(i)} = Y_1^{(i)} + dY_k^{(i)}$, where $dY_k^{(i)}$ is an approximate solution of the correction equation $A\,dY^{(i)} = BX^{(i)} - AY_1^{(i)}$. The stopping criterion requires that

$$\begin{aligned}\frac{\|BX^{(i)} - AY_{k+1}^{(i)}\|}{\|BX^{(i)}\|} &= \frac{\|BX^{(i)} - A(Y_1^{(i)} + dY_k^{(i)})\|}{\|BX^{(i)}\|} \\ &= \frac{\|(BX^{(i)} - AY_1^{(i)}) - A\,dY_k^{(i)}\|}{\|BX^{(i)} - AY_1^{(i)}\|}\,\frac{\|BX^{(i)} - AY_1^{(i)}\|}{\|BX^{(i)}\|} \leqslant \delta \sin\angle(A\mathcal{X}^{(i)}, B\mathcal{X}^{(i)}).\end{aligned} \tag{3.16}$$

Note that $\frac{\|(BX^{(i)} - AY_1^{(i)}) - A\,dY_k^{(i)}\|}{\|BX^{(i)} - AY_1^{(i)}\|}$ is the relative residual norm of the correction equation for $dY_k^{(i)}$, and $\frac{\|BX^{(i)} - AY_1^{(i)}\|}{\|BX^{(i)}\|}$ is the relative residual norm of the original equation for $Y_1^{(i)}$. It follows that the prescribed stopping criterion of the inner iteration is satisfied if $\frac{\|(BX^{(i)} - AY_1^{(i)}) - A\,dY_k^{(i)}\|}{\|BX^{(i)} - AY_1^{(i)}\|}$ is bounded above by

$$\frac{\delta\|BX^{(i)}\|\sin(A\mathcal{X}^{(i)}, B\mathcal{X}^{(i)})}{\|BX^{(i)} - AY_1^{(i)}\|} \geqslant \frac{\delta\|BX^{(i)}\|\sigma^{(i)}\mathrm{sep}(T_{22}, K^{(i)})}{\left\|\left(I - A(\mathbb{P}^{(i)})^{-1}\right)AV_2\right\| \|\overline{\mathcal{S}}_i\|\sqrt{1 + \|Q\|^2}} \equiv \rho^{(i)}, \tag{3.17}$$

where we apply the lower bound of $\sin(A\mathcal{X}^{(i)}, B\mathcal{X}^{(i)})$ in (2.14) and the upper bound of $\|BX^{(i)} - AY_1^{(i)}\|$ in (3.15).

To study $\rho^{(i)}$, recall from the end of the proof of Proposition 2.2 that $\sigma^{(i)} = \sigma_{min}(B)\|X^{(i)*}A^*AX^{(i)}\|^{-1/2} \to \sigma_{min}(B)\|V_1^*A^*AV_1\|^{-1/2} > 0$, and $\mathrm{sep}(T_{22}, K^{(i)}) \to \mathrm{sep}(T_{22}, K) > 0$. In addition, $\left\|\left(I - A(\mathbb{P}^{(i)})^{-1}\right)AV_2\right\| \to \left\|\left(I - A\mathbb{P}^{-1}\right)AV_2\right\|$ and $\|\overline{\mathcal{S}}_i\| \to \|\overline{\mathcal{S}}\|$. This means $\rho^{(i)}$, a lower bound of the relative tolerance for the correction equation, can be fixed independent of i. It then follows from Lemma 3.1 and our assumption concerning the decomposition (3.10) of $BX^{(i)} - AY_1^{(i)}$ that if block-GMRES is used to solve the correction equation, the inner iteration counts do not increase with the progress of the outer iteration. □

3.3. *A general strategy for the phase I computation*

It can be seen from Theorem 3.3 that the key to the success of the two-phase strategy is that in the first phase, an approximate solution $Y_1^{(i)}$ is obtained whose block residual norm $\|BX^{(i)} - AY_1^{(i)}\| = O(\|S^{(i)}\|)$. It is shown in Section 3.2 that such a $Y_1^{(i)}$ can be constructed inexpensively from a single step of block-GMRES with tuning applied to $AY^{(i)} = BX^{(i)}$. In fact, a valid $Y_1^{(i)}$ can also be constructed in other ways, in particular, by solving a set of least squares problems

$$\min_{f\in\mathbb{C}^p} \|BX^{(i)}e_j - AX^{(i)}f_j\| \qquad 1 \leqslant j \leqslant p. \tag{3.18}$$

This is easily done using the QR factorization of $AX^{(i)}$. The solution f_j satisfies

$$\begin{aligned}\|BX^{(i)}e_j - AX^{(i)}f_j\| &= \min_{f\in\mathbb{C}^p}\|BX^{(i)}e_j - AX^{(i)}f\| \\ &\leqslant \|BX^{(i)}e_j - AX^{(i)}(C^{(i)})^{-1}K^{-1}C^{(i)}e_j\| \\ &= \left\|AV_2\left(S^{(i)}(C^{(i)})^{-1}K^{-1}C^{(i)} - M^{-1}S^{(i)}\right)e_j\right\| \qquad \text{(see (2.9))} \\ &\leqslant \|AV_2\|\|\overline{S}_i\|\|S^{(i)}\| = O(\|S^{(i)}\|). \end{aligned} \tag{3.19}$$

Thus, with $Y_1^{(i)} = X^{(i)}[f_1, \ldots, f_p]$, it follows immediately that

$$\|BX^{(i)} - AY_1^{(i)}\| \leqslant \sum_{j=1}^{p}\|BX^{(i)}e_j - AX^{(i)}f_j\| \leqslant O(\|S^{(i)}\|), \tag{3.20}$$

so that the conclusion of Theorem 3.3 is also valid for this choice of $Y_1^{(i)}$.

This discussion reveals a connection between the two-phase strategy and the inverse correction method [29,17] and the residual inverse power method [34], where the authors independently present essentially the same key idea for inexact inverse iteration. For example [34], constructs $x^{(i+1)}$ by adding a small correction $z^{(i)}$ to $x^{(i)}$. Here, $z^{(i)}$ is the solution of $Az = \mu x^{(i)} - Ax^{(i)}$, where $\mu = x^{(i)*}Ax^{(i)}$ is the Rayleigh quotient, and $\mu x^{(i)} - Ax^{(i)}$ is the current eigenvalue residual vector that satisfies $\|\mu x^{(i)} - Ax^{(i)}\| = \min_{\alpha\in\mathbb{C}}\|\alpha x^{(i)} - Ax^{(i)}\|$. In Algorithm 2, we compute $Y_{k+1}^{(i)}$ by adding $dY_k^{(i)}$ to $Y_1^{(i)}$, where $dY_k^{(i)}$ is an approximate solution of $A\,dY^{(i)} = BX^{(i)} - AY_1^{(i)}$. Here $Y_1^{(i)}$ satisfies $\text{span}\{Y_1^{(i)}\} \approx \text{span}\{X^{(i)}\}$ (see (3.12)), and $\|BX^{(i)} - AY_1^{(i)}\|$ is minimized by a single block-GMRES iteration. For both methods, the relative tolerance of the correction equation can be fixed independent of the outer iteration. The least squares formulation derived from (3.18) can be viewed as a generalization to subspace iteration of the residual inverse power method of [34].

Remark 3.4. In fact, all these approaches are also similar to what is done by the Jacobi–Davidson method. To be specific, the methods in [29,17,34] essentially compute a parameter β explicitly or implicitly such that $\|B(\beta x^{(i)}) - Ax^{(i)}\|$ is minimized or close to being minimized, then solve the correction equation $Az^{(i)} = B(\beta x^{(i)}) - Ax^{(i)}$ and get $x^{(i+1)}$ by normalizing $x^{(i)} + z^{(i)}$. The right hand side $B(\beta x^{(i)}) - Ax^{(i)}$ is identical or similar to that of the Jacobi–Davidson correction equation, i.e., the current eigenvalue residual vector. The difference is that the system solve required by the Jacobi–Davidson method forces the correction direction to be orthogonal to the current approximate eigenvector $x^{(i)}$. In addition [14], shows that for inexact Rayleigh quotient iteration, solving the equation $(A - \sigma^{(i)}I)y^{(i)} = x^{(i)}$ ($\sigma^{(i)}$ is the Rayleigh quotient) with preconditioned full orthogonalization method (FOM) with *tuning* is equivalent to solving the simplified Jacobi–Davidson correction equation $(I - x^{(i)}x^{(i)*})(A - \sigma^{(i)}I)(I - x^{(i)}x^{(i)*})z^{(i)} = -(A - \sigma^{(i)})x^{(i)}$ with preconditioned FOM, as both approaches give the same inner iterate up to a constant.

4. Additional strategies to reduce inner iteration cost

In this section, we propose and study the use of deflation of converged Schur vectors, special starting vector for the correction equation, and iterative linear solvers with recycled subspaces to further reduce the cost of inner iteration.

4.1. Deflation of converged Schur vectors

With proper deflation of converged Schur vectors, we only need to apply matrix–vector products involving $\mathcal{A}$ to the unconverged Schur vectors. This reduces the inner iteration cost because the right hand side of the block linear system contains fewer columns. To successfully achieve this goal, two issues must be addressed: (1) how to simplify the procedure to detect converged Schur vectors and distinguish them from unconverged ones, and (2) how to apply tuning correctly to the block linear systems with reduced size, so that the relative tolerance of the correction equation can be fixed as in Theorem 3.3.

The first issue is handled by the Schur–Rayleigh–Ritz (SRR) procedure in Step 3 of Algorithm 1. The SRR step recombines and reorders the columns of $X^{(i+1)}$, so that its leading (leftmost) columns are approximate Schur vectors corresponding to the most dominant eigenvectors. Specifically, it forms the approximate Rayleigh quotient $\Theta^{(i)} = X^{(i)*}Y^{(i)} \approx X^{(i)*}\mathcal{A}X^{(i)}$, computes the Schur decomposition $\Theta^{(i)} = W^{(i)}T^{(i)}W^{(i)*}$ where the eigenvalues are arranged in *descending* order of magnitude in $T^{(i)}$, and orthogonalizes $Y^{(i)}W^{(i)}$ into $X^{(i+1)}$ (see Chapter 6 of [33]). As a result, the columns of $X^{(i+1)}$ will converge in order from left to right as the outer iteration proceeds. Then we only need to detect how many leading columns of $X^{(i+1)}$ have converged; the other columns are the unconverged Schur vectors.

To study the second issue, assume that $X^{(i)} = \left[X_a^{(i)}, X_b^{(i)}\right]$ where $X_a^{(i)}$ has converged. Then we deflate $X_a^{(i)}$ and solve the smaller block system $AY_b^{(i)} = BX_b^{(i)}$. When a single step of preconditioned block-GMRES with tuning is applied to this system (Phase I of Algorithm 2), it is important to *not* deflate $X_a^{(i)}$ in the tuned preconditioner (3.8). In particular, the effect of tuning (significant reduction of the linear residual norm in the first block-GMRES step) depends on the fact that $BX^{(i)}$ is an approximate invariant subspace of $A(\mathbb{P}^{(i)})^{-1}$. This nice property is valid only if we use the whole $X^{(i)}$ to define tuning.

To see this point, recall the partial Schur decomposition $B^{-1}AV_1 = V_1T_{11}$ in (2.2). We can further decompose this equality as

$$B^{-1}AV_1 = B^{-1}A\,[V_{1a},\ V_{1b}] = [V_{1a},\ V_{1b}] \begin{bmatrix} T_{11}^{\alpha} & T_{11}^{\beta} \\ 0 & T_{11}^{\gamma} \end{bmatrix} = V_1T_{11}. \tag{4.1}$$

It follows that $AV_{1b} = BV_{1a}T_{11}^{\beta} + BV_{1b}T_{11}^{\gamma}$, or equivalently

$$\left(-AV_{1a}(T_{11}^{\alpha})^{-1}T_{11}^{\beta} + AV_{1b}\right)(T_{11}^{\gamma})^{-1} = BV_{1b}. \tag{4.2}$$

In short, $\text{span}\{BV_{1b}\} \subset \text{span}\{AV_{1a}\} \cup \text{span}\{AV_{1b}\} = \text{span}\{AV_1\}$, but $\text{span}\{BV_{1b}\} \not\subseteq \text{span}\{AV_{1b}\}$, because of the triangular structure of the partial Schur form in (4.1). If $X_a^{(i)} = V_{1a}$ is the set of converged dominant Schur vectors, and $X_b^{(i)} \approx V_{1b}$ is the set of unconverged Schur vectors, then these observations show that $B\mathcal{X}_b^{(i)}$ has considerable components in *both* $A\mathcal{X}_a^{(i)}$ and $A\mathcal{X}_b^{(i)}$. Therefore, when solving $AY_b^{(i)} = BX_b^{(i)}$, if we use *only* $X_b^{(i)}$ to define tuning in (3.8), so that $A\mathcal{X}_a^{(i)}$ is *not* an invariant subspace of $A(\mathbb{P}^{(i)})^{-1}$, then the right hand side $BX_b^{(i)}$ does not span an approximate invariant subspace of $A(\mathbb{P}^{(i)})^{-1}$. Thus the large one-step reduction of the linear residual norm (see (3.15)) will not occur, and many more block-GMRES iterations would be needed for the correction equation.

4.2. Special starting vector for the correction equation

The second additional means to reduce the inner iteration cost is to choose a good starting vector for the correction equation, so that the initial residual norm of the correction equation can be greatly reduced. We find that a good starting vector for the current equation can be constructed from a proper linear combination of the solutions of previously solved equations, because the right hand sides of several consecutive correction equations are close to being linearly dependent. Note that the feasibility of this construction of starting vector stems from the specific structure of the two-phase strategy: as tuning defined in (3.8) need not be applied in Phase II of Algorithm 2, the preconditioner does not depend on $X^{(i)}$, and thus we can work with preconditioned system matrices that are the same for the correction equation in all outer iterations.

To understand the effectiveness of this special starting vector, we need to see why the right hand sides of a few successive correction equations are close to being linearly dependent. Some insight can be obtained by analyzing the simple case with block size $p = 1$. To begin the analysis, consider using Algorithm 2 to solve $Ay^{(i)} = Bx^{(i)}$, where $x^{(i)} = v_1 c^{(i)} + V_2 S^{(i)}$ is the normalized current approximate eigenvector (see (2.6)). Here $c^{(i)}$ is a scalar and $S^{(i)} \in \mathbb{C}^{(n-1)\times 1}$ is a column vector.

In Phase I of Algorithm 2, we apply a single step of preconditioned GMRES to solve $A(\mathbb{P}^{(i)})^{-1}\tilde{y}^{(i)} = Bx^{(i)}$ and get $y_1^{(i)} = (\mathbb{P}^{(i)})^{-1}\tilde{y}_1^{(i)}$. We know from the property of GMRES that $y_1^{(i)} = \alpha(\mathbb{P}^{(i)})^{-1}Bx^{(i)}$ where

$$\alpha = \operatorname{argmin}_\alpha \|Bx^{(i)} - \alpha A(\mathbb{P}^{(i)})^{-1}Bx^{(i)}\| = \frac{(Bx^{(i)})^* A(\mathbb{P}^{(i)})^{-1}Bx^{(i)}}{\|A(\mathbb{P}^{(i)})^{-1}Bx^{(i)}\|^2} = \frac{\nu_\alpha}{\mu_\alpha}. \tag{4.3}$$

To evaluate α, noting that $K = \lambda_1$ in (2.3), we have from (2.9) that

$$Bx^{(i)} = \lambda_1^{-1}Ax^{(i)} - AV_2(\lambda_1^{-1}I - M^{-1})S^{(i)}. \tag{4.4}$$

From the tuning condition (3.8), $A(\mathbb{P}^{(i)})^{-1}(Ax^{(i)}) = Ax^{(i)}$, and therefore

$$A(\mathbb{P}^{(i)})^{-1}Bx^{(i)} = \lambda_1^{-1}Ax^{(i)} - A(\mathbb{P}^{(i)})^{-1}AV_2(\lambda_1^{-1}I - M^{-1})S^{(i)}. \tag{4.5}$$

To simplify the notation, let $J_1 = AV_2(\lambda_1^{-1}I - M^{-1})$ in (4.4) and $J_2 = A(\mathbb{P}^{(i)})^{-1}J_1$ in (4.5). Substituting (4.4) and (4.5) into (4.3), we get the numerator and denominator of α as follows:

$$\nu_\alpha = \lambda_1^{-2}\|Ax^{(i)}\|^2 - \lambda_1^{-1}(Ax^{(i)})^*(J_1 + J_2)S^{(i)} + S^{(i)*}J_1^*J_2S^{(i)}, \tag{4.6}$$

and

$$\mu_\alpha = \lambda_1^{-2}\|Ax^{(i)}\|^2 - 2\lambda_1^{-1}(Ax^{(i)})^*J_2S^{(i)} + S^{(i)*}J_2^*J_2S^{(i)}. \tag{4.7}$$

Both ν_α and μ_α are quadratic functions of $S^{(i)}$. As $\|S^{(i)}\| \to 0$, we use the first order approximation of $\alpha = \nu_\alpha/\mu_\alpha$ (neglecting higher order terms in $S^{(i)}$) to get

$$\alpha \approx 1 + \frac{\lambda_1}{\|Ax^{(i)}\|^2}(Ax^{(i)})^*(J_2 - J_1)S^{(i)}. \tag{4.8}$$

Assume that $M = U_M\Lambda_M U_M^{-1}$ where $\Lambda_M = \operatorname{diag}(\lambda_2, \lambda_3, \ldots, \lambda_n)$ and M has normalized columns. It follows that $V_2U_M = [v_2, v_3, \ldots, v_n]$ are the normalized eigenvectors of (A, B) and $AV_2U_M = BV_2U_M\Lambda_M$. Suppose $x^{(0)} = c_1v_1 + \sum_{k=2}^n c_kv_k$. If the system $Ay^{(i)} = Bx^{(i)}$ is solved exactly for all i, then $x^{(i)} = c_1v_1 + \sum_{k=2}^n c_k(\lambda_1/\lambda_k)^i v_k$ up to some constant scaling factor. On the other hand, we know from (2.6) that $x^{(i)} = v_1c^{(i)} + V_2S^{(i)} = v_1c^{(i)} + V_2U_M(U_M^{-1}S^{(i)})$. Therefore,

$$U_M^{-1}S^{(i)} = \left(c_2\left(\frac{\lambda_1}{\lambda_2}\right)^i, c_3\left(\frac{\lambda_1}{\lambda_3}\right)^i, \ldots, c_n\left(\frac{\lambda_1}{\lambda_n}\right)^i\right)^* \tag{4.9}$$

up to some constant. If $\{|\lambda_2|, |\lambda_3|, \ldots, |\lambda_l|\}$ are tightly clustered and well-separated from $\{|\lambda_{l+1}|, \ldots, |\lambda_n|\}$, the magnitudes of the first $l-1$ entries of $U_M^{-1}S^{(i)}$ are significantly bigger than those of the other entries.

With (4.4), (4.5), (4.8) and (4.9), using the first order approximation again, we can write the right hand side of the correction equation as

$$\begin{aligned} Bx^{(i)} - Ay_1^{(i)} &= Bx^{(i)} - \alpha A(\mathbb{P}^{(i)})^{-1}Bx^{(i)} \\ &\approx \left(I - \frac{Ax^{(i)}}{\|Ax^{(i)}\|}\left(\frac{Ax^{(i)}}{\|Ax^{(i)}\|}\right)^*\right) \\ &\quad \times \left(A(\mathbb{P}^{(i)})^{-1} - I\right)A(V_2U_M)(\lambda_1^{-1}I - \Lambda_M^{-1})(U_M^{-1}S^{(i)}), \end{aligned} \tag{4.10}$$

where the vector $(V_2U_M)(\lambda_1^{-1}I - \Lambda_M^{-1})(U_M^{-1}S^{(i)})$ is dominated by a linear combination of $\{v_2, \ldots, v_l\}$. In addition, as $x^{(i)} \to v_1$, the first matrix factor in the second line of (4.10) filters out the component of Av_1. As a result, for big enough i, the right hand side of the current correction equation is roughly a linear combination of those of $l-1$ previous consecutive equations.

Using the above observation, a starting vector $dY_0^{(i)}$ for the correction equation can be constructed from l previous approximate solutions as follows:

$$dY_0^{(i)} = \mathrm{dY}_{l-1}y, \quad \text{where } \mathrm{dY}_{l-1} = \left[dY^{(i-l+1)}, dY^{(i-l+2)}, \ldots, dY^{(i-1)}\right], \text{ and} \tag{4.11}$$

$$y = \mathrm{argmin}_{y\in\mathbb{C}^{l-1}} \left\| \mathrm{RHS}_{l-1}y - \left(BX^{(i)} - AY_1^{(i)}\right)\right\|,$$

$$\text{where} \quad \mathrm{RHS}_{l-1} = \left[BX^{(i-l+1)} - AY_1^{(i-l+1)}, \ldots, BX^{(i-1)} - AY_1^{(i-1)}\right]. \tag{4.12}$$

In practice, we find that $l = 3$ or 4 is enough to generate a good starting vector. The cost of solving this small least squares problem (4.12) is negligible.

4.3. Linear solvers with recycled subspaces

In Phase II of Algorithm 2, we need to solve the correction equation $A\,dY^{(i)} = BX^{(i)} - AY_1^{(i)}$. The third strategy to speed up the inner iteration is to use linear solvers with subspace recycling to solve the sequence of correction equations. This methodology is specifically designed to efficiently solve a long sequence of *slowly-changing* linear systems. After the iterative solution of one linear system, a small set of vectors from the current subspace for the candidate solutions is carefully selected and the space spanned by these vectors is "recycled", i.e., used for the iterative solution of the next linear system. The cost of solving subsequent linear systems can usually be reduced by subspace recycling, because the iterative solver does not have to build the subspace for the candidate solution from scratch. A typical solver of this type is the Generalized Conjugate Residual with implicit inner Orthogonalization and Deflated Restarting (GCRO-DR) in [27], which was developed using ideas for the solvers with special truncation [6] and restarting [24] for a single linear system. See [25] for related work.

In [27], the preconditioned system matrix changes from one linear system to the next, and the recycled subspace taken from the previous system must be transformed by matrix–vector products involving the current system matrix to fit into the solution of the current system (see the Appendix of [27]). In the setting of solving the sequence of correction equations, fortunately, this transformation cost can be avoided with Algorithm 2, because the preconditioned system matrix is the same for the correction equation in all outer iterations.

We implement a block version of GCRO-DR to solve the correction equation. The block generalization is very similar to the extension of GMRES to block-GMRES. The residual norm of the block linear system is minimized in each block iteration over all candidate solutions in the union of the recycled subspace and a block Krylov subspace (see [27] for details). The dimension of the recycled subspace can be chosen independent of the block size. The authors of [27] suggest choosing the harmonic Ritz vectors corresponding to smallest harmonic Ritz values for the recycled subspaces. The harmonic Ritz vectors are approximate "smallest" eigenvectors of the preconditioned system matrix. If they do approximate these eigenvectors reasonably well, this choice tends to reduce the duration of the initial latency of GMRES convergence, which is typically observed when the system matrix has a few eigenvalues of very small magnitude; see [9]. We also include dominant Ritz vec tors in the recycled subspace, as suggested in [27]. As our numerical experiments show (see Section 5), when the use of harmonic Ritz vectors fails to reduce the inner iteration cost, the set of dominant Ritz vectors is still a reasonable choice for subspace recycling.

5. Numerical experiments

In this section, we test the effectiveness of the strategies described in Sections 3 and 4 for solving the block linear systems arising in inexact subspace iteration. We show that the two-phase strategy (Algorithm 2) achieves performance similar to that achieved when tuning is used at *every* block-GMRES step (the approach given in [27]): both methods keep the inner iteration cost from increasing, though the required tolerance for the solve decreases progressively. The numerical experiments also corroborate the analysis that a single block-GMRES iteration with tuning reduces the linear residual norm to a small quantity proportional to $\|S^{(i)}\|$, so that the relative tolerance of the correction equation remains a moderately small constant independent of $\|S^{(i)}\|$. We have also seen experimentally that the least squares strategy of Section 3.3 achieves the same effect. The Phase I step is somewhat more expensive using tuned preconditioned GMRES than the least squares approach, but for the problems we studied, the former approach required slightly fewer iterations in Phase II, and the total of inner iterations is about the same for the two methods. For the sake of brevity, we only present the results obtained by the two-phase strategy where tuning is applied in Phase I.

We also show that deflation gradually decreases the inner iteration cost as more converged Schur vectors are deflated. In addition, the use of subspace recycling and special starting vector lead to further reduction of inner iteration counts.

We first briefly explain the criterion to detect the convergence of Schur vectors in Step 3 of Algorithm 1. Let $I_{p,j} = (I_j \ 0)^T \in \mathbb{R}^{p\times j}$ so that $X^{(i)}I_{p,j}$ contains the first j columns of $X^{(i)}$. Right after the SRR step, we find the largest integer j for which the following criterion is satisfied:

$$\|BX^{(i)}I_{p,j} - AX^{(i)}I_{p,j}T_j^{(i)}\| \leqslant \|BX^{(i)}I_{p,j}\|\epsilon, \tag{5.1}$$

where $T_j^{(i)}$ is the $j \times j$ leading block of $T^{(i)}$ coming from the SRR step (see Section 4.1 for details). If (5.1) holds for j but not for $j+1$, we conclude that exactly j Schur vectors have converged and should be deflated. This stopping criterion is analogous to that of the EB12 function (subspace iteration) of HSL (formerly the Harwell Subroutine Library) [19,21].

We use four test problems. The first one is MHD4800A/B from Matrix Market [22], a real matrix pencil of order 4800 which describes the Alfvén spectra in magnetohydrodynamics (MHD). We use the shift-invert operator $\mathcal{A} = (A - \sigma B)^{-1}B$ with σ close to the left end of the spectrum. Since it is very hard to find a preconditioner for A, we use the ILU preconditioner for $A - \sigma B$ with drop tolerance 1.5×10^{-7} given by MATLAB's `ilu`. Using MATLAB's `nnz` to count the number of nonzero entries, we have $\mathtt{nnz}(A - \sigma B) = 120{,}195$, and $\mathtt{nnz}(L) + \mathtt{nnz}(U) = 224{,}084$. In fact, a slightly bigger tolerance, say 1.75×10^{-7}, leads to failure of `ilu` due to a zero pivot.

The second problem is UTM1700A/B from Matrix Market, a real matrix pencil of size 1700 arising from a tokamak model in plasma physics. We use Cayley transformation to compute the leftmost eigenvalues $\lambda_{1,2} = -0.032735 \pm 0.3347i$ and $\lambda_3 = 0.032428$. Note that $\Im(\lambda_{1,2})$ is 10 times bigger than λ_3, and there are some real eigenvalues to the right of λ_3 with magnitude smaller than $\Im(\lambda_{1,2})$. We choose the ILU preconditioner with drop tolerance 0.001 for $A - \sigma_1 B$.

Problems 3 and 4 come from the linear stability analysis of a model of two-dimensional incompressible fluid flow over a backward facing step, constructed using the IFISS software package [7,8]. The domain is $[-1, L] \times [-1, 1]$, where $L = 15$ in Problem 3 and $L = 22$ in Problem 4; the Reynolds numbers are 600 and 1200, respectively. Let u and v be the horizontal and vertical components of the velocity, p be the pressure, and ν the viscosity. The boundary conditions are as follows:

$$\begin{aligned}
&u = 4y(1-y),\ v = 0 \text{ (parabolic inflow)} && \text{on } x = -1, y \in [0,1];\\
&\nu\frac{\partial u}{\partial x} - p = 0,\ \frac{\partial v}{\partial y} = 0 \text{ (natural outflow)} && \text{on } x = L, y \in [-1,1];\\
&u = v = 0 \text{ (no-slip)} && \text{on all other boundaries.}
\end{aligned} \tag{5.2}$$

We use a biquadratic/bilinear $(Q_2 - Q_1)$ finite element discretization with element width $\frac{1}{16}$ (grid parameter 6 in the IFISS code). The sizes of the two problems are 72,867 and 105,683, respectively. Block linear solves are done using the least squares commutator preconditioner [10]. For Problems 3 and 4, we try both shift-invert (subproblem (a)) and Cayley transformation (subproblem (b)) to detect a small number of critical eigenvalues.

For completeness, we summarize all parameters used in the solution of each test problem in Table 1. These parameters are chosen to deliver approximate eigenpairs of adequate accuracies, show representative behavior of each solution strategy, and keep the total computational cost moderate.

Table 1
Parameters used to solve the test problems.

	p	k	σ (σ_1)	σ_2	δ	ϵ	l_1	l_2
Problem 1	9	7	-370	–	2×10^{-5}	5×10^{-11}	5	10
Problem 2	3	3	-0.0325	0.125	1×10^{-5}	5×10^{-11}	5	10
Problem 3(a)	7	7	0	–	1×10^{-3}	5×10^{-10}	0	20
Problem 3(b)	5	3	0	-0.46	1×10^{-4}	5×10^{-10}	0	20
Problem 4(a)	5	4	0	–	1×10^{-3}	5×10^{-10}	0	30
Problem 4(b)	4	4	0	-0.24	5×10^{-4}	5×10^{-10}	0	30

1. p, k – we use $X^{(i)}$ with p columns to compute k eigenpairs of (A, B).
2. $\sigma, \sigma_1, \sigma_2$ – the shifts of $\mathcal{A} = (A - \sigma B)^{-1}B$ and $\mathcal{A} = (A - \sigma_1 B)^{-1}(A - \sigma_2 B)$.
3. δ – the relative tolerance for solving $AY^{(i)} = BX^{(i)}$ is $\delta \sin \angle(A\mathcal{X}^{(i)}, B\mathcal{X}^{(i)})$.
4. ϵ – the error in the convergence test (5.1).
5. l_1, l_2 – we use l_1 "smallest" harmonic Ritz vectors and l_2 dominant Ritz vectors for subspace recycling.

The performance of different strategies to solve $AY^{(i)} = BX^{(i)}$ for each problem is shown in Figs. 1–4. We use Problem 1 as an example to explain the results. In Fig. 1a, the preconditioned matrix–vector product counts of the inner iteration are plotted against the progress of the outer iteration. The curves with different markers correspond to solution strategies as follows:

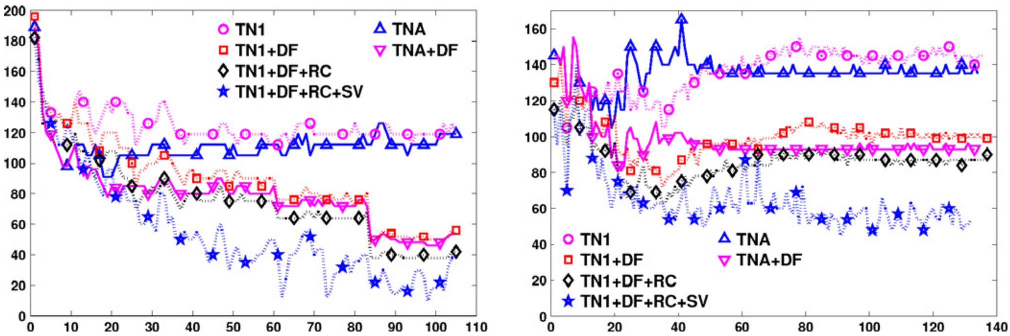

Fig. 3. Performance of different solution strategies for Problems 3(a) and 3(b): preconditioned matrix–vector product counts of the inner iteration against the outer iteration.

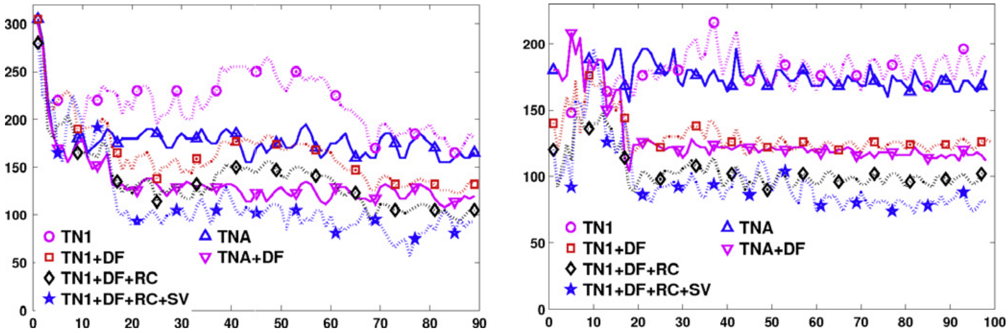

Fig. 4. Performance of different solution strategies for Problems 4(a) and 4(b): preconditioned matrix–vector product counts of the inner iteration against the outer iteration.

1. “NO-TN” (no marker with dotted line) – Solve $AY^{(i)} = BX^{(i)}$ by preconditioned block-GMRES *without* tuning.
2. “TNA” (△ marker with solid line) – Solve $AY^{(i)} = BX^{(i)}$ by preconditioned block-GMRES *with* tuning.
3. “TNA+DF” (▽ marker with solid line) – Apply “TNA” and deflation of converged Schur vectors.
4. “TN1” (○ marker with dashed line) – Solve $AY^{(i)} = BX^{(i)}$ by Algorithm 2, without any additional enhancements.
5. “TN1+DF” (□ marker with dashed line) – Apply “TN1” and deflation of converged Schur vectors.
6. “TN1+DF+RC” (◇ marker with dashed line) – Apply “TN1+DF” and use GCRO-DR to solve the correction equation in Phase II of Algorithm 2.
7. “TN1+DF+RC+SV” (★ marker wish dashed line) – Apply “TN1+DF+RC” and use the special starting vector for the correction equation.

From Fig. 1a, we see that if no tuning is used, the matrix–vector product counts in each outer iteration increase gradually to over 160, whereas they are fixed at 110 if the two-phase strategy (without any additional enhancements) is applied. If converged Schur vectors are deflated, the matrix–vector product counts decrease gradually from 100 to 80, 70, and finally to about 50. The use of recycled subspace of dimension 15 further reduces the counts by approximately 15. The special starting vector makes an additional significant improvement: the counts are reduced to less than 20 after the 23rd outer iteration, and even to less than 10 after the 35th outer iteration.

Fig. 1b plots four quantities against the outer iteration as follows:

1. “Tol-OrigEqn” (○ marker) – The relative tolerance $\delta \sin(A\mathcal{X}^{(i)}, B\mathcal{X}^{(i)})$ for the original linear system $AY^{(i)} = BX^{(i)}$.
2. “Res-OneStep” (◇ marker) – The relative residual norm $\frac{\|BX^{(i)}-AY_1^{(i)}\|}{\|BX^{(i)}\|}$ after one step block-GMRES iteration with tuning (Phase I of Algorithm 2).
3. “Tol-CrtEqn” (□ marker) – The relative tolerance $\frac{\delta\|BX^{(i)}\| \sin(A\mathcal{X}^{(i)},B\mathcal{X}^{(i)})}{\|BX^{(i)}-AY_1^{(i)}\|}$ for the correction equation $A\,dY^{(i)} = BX^{(i)} - AY_1^{(i)}$.
4. “Res-CrtEqn-SV” (△ marker) – Given the starting vector $dY_0^{(i)}$, the initial relative residual norm $\frac{\|(BX^{(i)}-AY_1^{(i)})-A\,dY_0^{(i)}\|}{\|BX^{(i)}-AY_1^{(i)}\|}$ of the correction equation.

It is clear from Fig. 1b that one step of block-GMRES with tuning reduces the residual norm of $AY^{(i)} = BX^{(i)}$ to a small quantity proportional to $\|S^{(i)}\|$ (“Res-OneStep”), so that the relative tolerance of the correction equation (“Tol-CrtEqn”) is approximately a constant. In addition, the special starting vector $dY_0^{(i)}$ considerably reduces the initial residual norm of the correction equation. For example, in the 45th outer iteration, the relative tolerance for $AY^{(45)} = BX^{(45)}$ is $\delta \sin(A\mathcal{X}^{(45)}, B\mathcal{X}^{(45)}) \approx 10^{-10}$; a single block-GMRES iteration with tuning decreases the relative linear residual norm to $\frac{\|BX^{(45)}-AY_1^{(45)}\|}{\|BX^{(45)}\|} \approx 10^{-6}$, so that $dY_k^{(45)}$ for the correction equation only needs to satisfy $\frac{\|(BX^{(45)}-AY_1^{(45)})-A\,dY_k^{(45)}\|}{\|BX^{(45)}-AY_1^{(45)}\|} \leqslant 10^{-4}$; see (3.16) for details. Moreover, the starting vector $dY_0^{(45)}$ makes $\frac{\|(BX^{(45)}-AY_1^{(45)})-A\,dY_0^{(45)}\|}{\|BX^{(45)}-AY_1^{(45)}\|}$ almost as small as 10^{-4}, so that little additional effort is needed to solve the correction equation.

Table 2 shows the number of preconditioned matrix–vector products when different strategies are used to solve the block linear systems $AY^{(i)} = BX^{(i)}$ for each problem. For Problems 1 and 2, we achieve a speed up ratio of 3.4 and 4.4, respectively, by the combined use of all strategies, compared to the original use of tuning (“TNA”) proposed in [28]; for Problems 3(a) and 3(b), we reduce the inner iteration cost by over 50%; for Problems 4(a) and 4(b), the savings are 36% and 45%, respectively. Recall

Table 2
Number of preconditioned matrix–vector products for different solution strategy for each problem.

	NO-TN	TNA	TN1	TNA + DF	TN1 + DF	TN1 + DF + RC	TN1 + DF + RC + SV
Problem 1	6435	3942	4761	2606	3254	2498	1175
Problem 2	19,110	12,183	15,357	10,854	13,886	7220	2744
Problem 3(a)	–	11,704	13,097	8270	9370	7863	5785
Problem 3(b)	–	17,475	18,600	12,613	13,792	11,806	8521
Problem 4(a)	–	15,785	19,350	11,978	14,578	12,183	10,100
Problem 4(b)	–	17,238	17,468	12,624	12,892	10,197	9428

from (3.9) that tuning requires an application of the Sherman–Morrison–Woodbury formula in each *inner* iteration. The two-phase strategy uses tuning only in one block-GMRES iteration and hence avoids the overhead of tuning. The additional strategies of Section 4 only entail computation of the recycled subspaces and the starting vector (both costs are small) for the block system in each *outer* iteration.

One can see from Figs. 1–4 that the two-phase strategy without subspace recycling and special starting vector generally requires slightly more inner iterations than the original tuned version of the solves (compare “TN1” with “TNA” and “TN1+DF” with “TNA+DF”). The reason is that the tuned version of a preconditioner $\mathbb{P}^{(i)}$ has two possible advantages over its untuned version P:

1. With a tuned preconditioner, the right hand side of $A(\mathbb{P}^{(i)})^{-1}\tilde{Y}^{(i)} = BX^{(i)}$ is an approximate invariant subspace of the preconditioned operator $A(\mathbb{P}^{(i)})^{-1}$.
2. In addition, $A(\mathbb{P}^{(i)})^{-1}$ typically has more favorable properties, such as better eigenvalue clustering, for Krylov subspace methods than AP^{-1}.

The first advantage is the original motivation for the use of tuning, as studied in [13,14,28] and this paper. The second one is studied in [15] for solving linear systems that arise when inexact Arnoldi method is applied to compute a few smallest eigenvalues of a matrix from Matrix Market. We attribute the slight increase in inner iteration counts associated with Algorithm 2 to its use of untuned preconditioners in the second phase. However, with Algorithm 2, the overhead of tuning is avoided, and further reduction of inner iteration counts can be achieved by using subspace recycling (no transformation of subspaces needed) and special starting vectors.

Moreover, our experience suggests that the second advantage of tuning tends to be less of a factor if the untuned preconditioner P is very strong (most eigenvalues of AP^{-1} are clustered around 1). For instance, for Problem 1, compared to the strategy “TNA” where tuning is used in every inner iteration, the two-phase strategy “TN1” requires about 18 more preconditioned matrix–vector products (or a 20% relative increase) for each block linear system after the 20th outer iteration; see Fig. 1a. Similarly for Problem 2, “TN1” needs about 15 more matrix–vector multiplications (or a 25% relative increase) than “TNA” for each system after the 75th outer iteration. However, for Problems 3(a), 3(b) and 4(b), the relative increase is only about 10% in the last tens of outer iterations; for Problem 4(a), though “TNA” obviously outperforms “TN1” in the first 67 outer iterations, the relative difference between the two approaches still falls far below 20% in the last 22 outer iterations. The reason is that the “clustering” effect of tuning is more pronounced when the relatively weak ILU preconditioners are used in Problems 1 and 2, and is less influential for Problems 3 and 4 where the strong least squares commutator preconditioner [9] is used.

In all numerical experiments, deflation of converged Schur vectors always reduces the *preconditioned matrix–vector product* counts, but the inner *iteration* counts tends to increase slightly. This agrees with our experience with the behavior of block linear solvers. For instance, if it takes 10 block iterations to solve a block system with 8 right hand sides to some tolerance, then it usually takes more than 10 but fewer than 20 block iterations to solve the system with block size 4 to the some tolerance.

We successfully reduce some inner iteration cost by using block GCRO-DR (subspace recycling). However, a conclusive evaluation of the effectiveness of this approach is beyond the scope of this paper. To the best of our knowledge, block GCRO-DR has not been mentioned in the literature. The dimensions of the recycled subspaces we use in block GCRO-DR are commensurate with those used in single-vector GCRO-DR [27]. Since block GCRO-DR generally needs much bigger subspaces to extract candidate solutions than its single-vector counterpart, it might be beneficial to use recycled subspaces of bigger dimensions. In addition, the harmonic Ritz vectors corresponding to smallest harmonic Ritz values are not necessarily a good choice for recycling if, for example, the smallest eigenvalues of the preconditioned system matrix are not well-separated from other eigenvalues [5]. We speculate this is the case in Problems 3 and 4, where there are several very small eigenvalues and some small ones when the least squares commutator preconditioner is used (see [9]). In this case, it is the dominant Ritz vectors that are useful.

6. Conclusion

We have studied inexact subspace iteration for solving generalized non-Hermitian eigenvalue problems with shift-invert and Cayley transformations. We provide new perspectives on tuning and discuss the connection of the two-phase strategy to the inverse correction method, the residual inverse power method and the Jacobi–Davidson method. The two-phase strategy applies tuning only in the first block-GMRES iteration and solves the correction equation with a fixed relative tolerance. It prevents the inner iteration counts from increasing as the outer iteration proceeds, as the original approach in [28] does. Three additional strategies are studied to further reduce the inner iteration cost, including deflation, subspace recycling and special initial guess. Numerical experiments show clearly that the combined use of all these techniques leads to significant reduction of inner iteration counts.

Acknowledgement

We thank the reviewers for a very careful reading of the manuscript.

References

[1] J. Berns-Müller, A. Spence, Inexact Inverse Iteration and GMRES, Tech Report maths0507, University of Bath, Bath, UK, 2005.
[2] J. Berns-Müller, A. Spence, Inexact inverse iteration with variable shift for nonsymmetric generalized eigenvalue problems, SIAM J. Matrix Anal. Appl. 28 (4) (2006) 1069–1082.
[3] J. Berns-Müller, I.G. Graham, A. Spence, Inexact inverse iteration for symmetric matrices, Linear Algebra Appl. 416 (2–3) (2006) 389–413.
[4] K.A. Cliffe, T.J. Garratt, A. Spence, Eigenvalues of block matrices arising from problems in fluid mechanics, SIAM J. Matrix Anal. 15 (4) (1994) 1310–1318.
[5] E. de Sturler, Private communication, October 2008.
[6] E. de Sturler, Truncation strategies for optimal Krylov subspace methods, SIAM J. Numer. Anal. 36 (3) (1999) 864–889.
[7] H.C. Elman, A.R. Ramage, D.J. Silvester, A.J. Wathen, Incompressible Flow Iterative Solution Software Package. Available from: <http://www.cs.umd.edu/elman/ifiss.html>.
[8] H.C. Elman, A.R. Ramage, D.J. Silvester, IFISS: a Matlab toolbox for modelling incompressible flow,ACM Trans. Math. Software 33 (2) (2007) 18, Article 14.
[9] H.C. Elman, D.J. Silvester, A.J. Wathen, Performance and analysis of saddle point preconditioners for the discrete steady-state Navier–Stokes equations, Numer. Math. 90 (4) (2002) 665–688.
[10] H.C. Elman, D.J. Silvester, A.J. Wathen, Finite Elements and Fast Iterative Solvers, Oxford University Press, New York, 2005.
[11] O.G. Ernst, Residual-minimizing Krylov subspace methods for stabilized discretizations of convection–diffusion equations, SIAM J. Matrix Anal. Appl. 21 (4) (2000) 1079–1101.
[12] M. Freitag, A. Spence, A tuned preconditioner for inexact inverse iteration applied to Hermitian eigenvalue problems, IMA J. Numer. Anal. 28 (3) (2007) 522–551.
[13] M. Freitag, A. Spence, Convergence rates for inexact inverse iteration with application to preconditioned iterative solves, BIT Numer. Math. 47 (1) (2007) 27–44.
[14] M. Freitag, A. Spence, Rayleigh quotient iteration and simplified Jacobi–Davidson method with preconditioned iterative solves, Linear Algebra Appl. 428 (8–9) (2008) 2049–2060.
[15] M. Freitag, A. Spence, Shift-invert Arnoldi's method with preconditioned iterative solves, SIAM J. Matrix Anal. Appl. 31 (3) (2009) 942–969.
[16] G.H. Golub, C.F. van Loan, Matrix Computations, third ed., The Johns Hopskins University Press, Baltimore, 1996.
[17] G. Golub, Q. Ye, Inexact inverse iteration for generalized eigenvalue problems, BIT Numer. Math. 40 (4) (2000) 671–684.
[18] M. Hochbruck, C. Lubich, On Krylov subspace approximations to the matrix exponential operator, SIAM J. Numer. Anal. 34 (5) (1997) 1911–1925.
[19] HSL. Available from: <http://www.cse.scitech.ac.uk/nag/hsl/>.
[20] Y. Lai, K. Lin, W. Lin, An inexact inverse iteration for large sparse eigenvalue problems, Numer. Linear Algebra Appl. 4 (5) (1997) 425–437.
[21] R.B. Lehoucq, J.A. Scott, An Evaluation of Subspace Iteration Software for Sparse Nonsymmetric Eigenvalue Problems, RAL-TR-96-022, Rutherford Appleton Lab., 1996.

[22] Matrix Market. Available from: <http://math.nist.gov/MatrixMarket/>.
[23] K. Meerbergen, A. Spence, D. Roose, Shift-invert and Cayley transforms for detection of rightmost eigenvalues of nonsymmetric matrices, BIT Numer. Math. 34 (3) (1994) 409–423.
[24] R.B. Morgan, GMRES with deflated restarting, SIAM J. Sci. Comput. 24 (1) (2002) 20–37.
[25] R.B. Morgan, Restarted block-GMRES with deflation of eigenvalues, Appl. Numer. Math. 54 (2) (2005) 222–236.
[26] Y. Notay, Convergence analysis of inexact Rayleigh quotient iteration, SIAM J. Matrix Anal. Appl. 24 (3) (2003) 627–644.
[27] M.L. Parks, E. De Sturler, G. Mackey, D.D. Johnson, S. Maiti, Recycling Krylov subspaces for sequences of linear systems, SIAM J. Sci. Comput. 28 (5) (2006) 1651–1674.
[28] M. Robbé, M. Sadkane, A. Spence, Inexact inverse subspace iteration with preconditioning applied to non-Hermitian eigenvalue problems, SIAM J. Matrix Anal. Appl. 31 (1) (2009) 92–113.
[29] U. Rüde, W. Schmid, Inverse Multigrid Correction for Generalized Eigenvalue Computations, Technical Report 338, Universität Augsburg, 1995.
[30] Y. Saad, Iterative Methods for Sparse Linear Systems, second ed., SIAM, Philadelphia, 2003.
[31] V. Simoncini, L. Eldén, Inexact Rayleigh quotient-type methods for eigenvalue computations, BIT Numer. Math. 42 (1) (2002) 159–182.
[32] P. Smit, M.H.C. Paardekooper, The effects of inexact solvers in algorithms for symmetric eigenvalue problems,Linear Algebra Appl. 287 (1999) 337–357.
[33] G.W. Stewart, Matrix Algorithms Volumn II: Eigensystems, SIAM, Philadelphia, 2001.
[34] G.W. Stewart, A Residual Inverse Power Method, UMIACS TR-2007-09, CMSC TR-4890, University of Maryland, 2007.