SIAM J. MATRIX ANAL. APPL.
Vol. 33, No. 2, pp. 433–459

FAST INEXACT IMPLICITLY RESTARTED ARNOLDI METHOD FOR GENERALIZED EIGENVALUE PROBLEMS WITH SPECTRAL TRANSFORMATION*

FEI XUE† AND HOWARD C. ELMAN‡

Abstract. We study an inexact implicitly restarted Arnoldi (IRA) method for computing a few eigenpairs of generalized non-Hermitian eigenvalue problems with spectral transformation, where in each Arnoldi step (outer iteration) the matrix-vector product involving the transformed operator is performed by iterative solution (inner iteration) of the corresponding linear system of equations. We provide new perspectives and analysis of two major strategies that help reduce the inner iteration cost: a special type of preconditioner with "tuning," and gradually relaxed tolerances for the solution of the linear systems. We study a new tuning strategy constructed from vectors in both previous and current IRA cycles, and we show how tuning is used in a new two-phase algorithm to greatly reduce inner iteration counts. We give an upper bound of the allowable tolerances of the linear systems and propose an alternative estimate of the tolerances. In addition, the inner iteration cost can be further reduced through the use of subspace recycling with iterative linear solvers. The effectiveness of these strategies is demonstrated by numerical experiments.

Key words. inexact implicitly restarted Arnoldi method, tuning, relaxation, subspace recycling

AMS subject classifications. 65F18, 65F15, 65H17, 65F10

DOI. 10.1137/100786599

1. Introduction.

Many scientific and engineering applications require a small group of eigenvalues closest to a specified shift or those with largest or smallest real parts. The shift-invert and Cayley transformations [24] are the two most commonly used spectral transformations to map these eigenvalues to the dominant ones of the transformed operator, so that they can be readily computed by eigenvalue algorithms. The major challenge of this approach is that a linear system of equations involving a shifted matrix needs to be solved in each step (outer iteration) of the eigenvalue algorithm. For large-scale applications, for instance, finite element discretization of three-dimensional partial differential equations, this linear solve has to be done using iterative solvers (inner iteration) instead of factorization-based direct solvers. This offers the prospect of inexact eigenvalue algorithms with "inner-outer" structure, where the required solution of linear systems is computed only to a specified accuracy. This paper concerns efficient iterative solution of the linear systems of equations that arise when the inexact implicitly restarted Arnoldi (IRA) method with spectral transformation is used to detect a few eigenpairs of generalized non-Hermitian eigenvalue problems (GNHEP) $Av = \lambda Bv$.

In the past decade, considerable progress has been made in understanding inexact eigenvalue algorithms, especially the simplest one—inexact inverse iteration. Systematic study of this algorithm is mainly carried out by Spence and his collabo-

*Received by the editors February 22, 2010; accepted for publication (in revised form) by G. L. G. Sleijpen January 25, 2012; published electronically June 5, 2012. This work was supported by the U.S. Department of Energy under grants DEFG0204ER25619 and DEFG0205ER25672 and by the U.S. National Science Foundation under grants CCF0726017 and DMS-1115520.
http://www.siam.org/journals/simax/33-2/78659.html
†Department of Mathematics, Temple University, Philadelphia, PA 19122 (fxue@temple.edu).
‡Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742 (elman@cs.umd.edu).

rators (see [1, 2, 3, 13, 14, 15]). A major concern in these papers is the connection between the error of the inner solve and the convergence of the outer iteration, with different choices of variable shifts, tolerances, and formulations of the linear systems. Meanwhile, there has been increasing interest in reducing the inner iteration cost to enhance the effectiveness of the algorithm. Reference [29] gives some new insights into preconditioning the linear systems arising in inexact Rayleigh quotient iteration by modifying the right-hand side of the preconditioned system. This idea is extended in [1, 2, 3] and further refined in [14, 15] for inexact inverse iteration or Rayleigh quotient iteration, where a special type of preconditioner with "tuning" is constructed and analyzed. In [27, 42], tuning is used in the iterative solution of the block linear systems arising in inexact subspace iteration. In all these algorithms, tuning makes the preconditioned right-hand side of the linear system an approximate eigenvector (or invariant subspace) of the preconditioned system matrix, and hence the inner iteration counts are considerably reduced. This motivation of tuning has also recently been shown in [16] to bear an interesting relation to the Jacobi–Davidson method.

In the meantime, some developments have been made in understanding inexact projection-based eigenvalue algorithms, such as the Lanczos and the Arnoldi methods. It was found in [20] and [4] that the matrix-vector product must be computed accurately in the initial Lanczos or Arnoldi steps, but the accuracy can be relaxed as the algorithm proceeds without obviously affecting the convergence of approximate eigenpairs. An analysis of this phenomenon is given in [28] for the Arnoldi method, using perturbation theory of invariant subspaces. It is shown there that the allowable errors of matrix-vector products in Arnoldi steps should be inversely proportional to the eigenvalue residual norm of the desired eigenpair. Therefore, as the Arnoldi method proceeds and converges to the eigenpair of interest, the accuracy of matrix-vector products can be relaxed. The use of inexact matrix-vector products has also been studied in the setting of Krylov subspace linear solvers; see [5, 6], [30, 31, 32], and [40, 41].

Further study of inexact Arnoldi methods is given in [17], where the tuning strategy and the relaxed accuracy of matrix-vector products are extended to inexact IRA with shift-invert transformation for standard eigenvalue problems. For the linear systems arising in Arnoldi steps (outer iterations) in a given IRA cycle, tuning is developed using all available Arnoldi vectors in that cycle. Numerical experiments show that for a test problem from Matrix Market [23], an ILU preconditioner with this tuning considerably reduces the inner iteration counts. It is observed there and confirmed in this paper that this improvement is mainly due to the fact that tuning helps cluster the eigenvalues of the preconditioned system matrix of the linear system in each Arnoldi step. In addition, [17] proposes a practical estimate of the allowable relaxed tolerances for the solution of the linear systems, using the distance between the spectra of two matrices containing the wanted and unwanted Ritz values; this is proposed as a simpler alternative to the separation between the two matrices [36]. Numerical experiments show that the total inner iteration counts of inexact IRA can be substantially reduced by the combined use of tuning and relaxed tolerances.

In this paper, we refine the tuning strategy and further study the allowable tolerances for inner solves of the inexact IRA method for generalized non-Hermitian eigenvalue problems. We first study a new tuning strategy constructed for a given Arnoldi step using the solutions of linear systems obtained in previous Arnoldi steps. In addition, we propose a two-phase strategy to solve the linear system in the current Arnoldi step. Specifically, we first use tuning to obtain a minimum residual solution in the direction of the right-hand side of the current linear system, then solve a correction equation with any appropriate preconditioned linear solver; in particular, tuning

is not needed for the correction equation. We show that the approximate solution obtained in the first phase can be a very good one if enough solution vectors from previous Arnoldi steps are used. With this special approximate solution, the correction equation can be solved with a relative tolerance much larger than that of the original linear system, and inner iteration counts can hence be reduced considerably. In addition, we use a special type of iterative linear solver with subspace recycling to solve the sequence of correction equations as the IRA method proceeds. We show that subspace recycling is cheap to use in this setting and can further reduce the inner iteration counts substantially.

The second goal of this paper is to present a refined analysis of the allowable tolerance for the linear systems in the inexact IRA method. We first give an upper bound of the allowable tolerance, showing that violation of this bound necessarily leads to contamination of the desired approximate invariant subspace by excessive errors from the inner solves. We then give a theoretically more accurate estimate of the allowable tolerance, which is between the upper bound and a conservative lower bound from [17]. As this estimate contains information not available until the end of the current IRA cycle, we use a computable substitute obtained at the end of the previous IRA cycle. We then compare this heuristic estimate with that from [17] and discuss the impact of the accuracy of the estimate on the inner solves.

This paper is organized as follows. In section 2, we briefly review spectral transformations, the IRA method and some properties of the algorithm when exact shifts (unwanted Ritz values) are used in filter polynomials. We discuss a few strategies for the inner solves in section 3, studying the properties of the new tuning strategy and the new two-phase strategy for solving the linear system in each Arnoldi step. We also explain the effectiveness of the linear solver with subspace recycling applied to solve the correction equations. In section 4, we study the allowable tolerances of the linear systems and give a necessary upper bound for the tolerance. A new heuristic estimate of the allowable tolerance is proposed and used in numerical experiments to corroborate the accuracy of the estimate from [17]. Numerical experiments in section 5 show that the combined use of the new tuning, subspace recycling, and relaxed tolerances greatly reduces the total inner iteration counts. In section 6 we make some concluding remarks.

2. Review: Spectral transformations and the IRA method. To make the exposition smooth, we briefly review two commonly used spectral transformations and the IRA method.

The shift-invert and generalized Cayley transformations (see [24]) are usually used to detect interior eigenvalues or ones with large imaginary parts. They are defined as follows:

$$Av = \lambda Bv \;\Leftrightarrow\; (A-\sigma B)^{-1}Bv = \left(\frac{1}{\lambda-\sigma}\right)v \quad \text{(shift-invert)},$$

$$Av = \lambda Bv \;\Leftrightarrow\; (A-\sigma_1 B)^{-1}(A-\sigma_2 B)v = \left(\frac{\lambda-\sigma_2}{\lambda-\sigma_1}\right)v \quad \text{(generalized Cayley)}. \tag{2.1}$$

The shift-invert transformation maps eigenvalues near σ to dominant eigenvalues of $\mathcal{A} = (A-\sigma B)^{-1}B$; the 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 dominant eigenvalues of $\mathcal{A}$ can then be found by iterative eigenvalue algorithms. Once the eigenvalues of the transformed problem are obtained, they are

transformed back to those of the original problem; the eigenvectors do not change with the transformation.

Without loss of generality, we use the generic notation $\mathcal{A} = A^{-1}B$ for which we seek the k eigenvalues of $Av = \lambda Bv$ with smallest magnitude (i.e., k dominant eigenvalues of $\mathcal{A}$). This notation, in principle, covers both types of operators in (2.1) with any shifts. For example, let $\widehat{A} = A - \sigma_1 B$ and $\widehat{B} = A - \sigma_2 B$, so that the Cayley operator is $\mathcal{A} = \widehat{A}^{-1}\widehat{B}$. This generic notation $\mathcal{A} = A^{-1}B$ is used throughout this paper, unless otherwise stated.

Both the shift-invert and the Cayley transformations have been implemented in ARPACK [22], a mathematical software package of high quality which has become the standard solver for large non-Hermitian eigenvalue problems. ARPACK is based on the implicitly restarted Arnoldi (IRA) method, a well-known and important method for eigenvalue computation developed by Sorensen [34] in 1992.

The key technique of the IRA method is the implicit application of a filter polynomial to a given Arnoldi decomposition to produce the effect of several steps of a restarted Arnoldi computation without any matrix-vector multiplications. Specifically, at the end of the ith IRA cycle we have an m-step Arnoldi decomposition

$$\mathcal{A}U_m^{(i)} = U_m^{(i)} H_m^{(i)} + h_{m+1,m}^{(i)} u_{m+1}^{(i)} e_m^T. \tag{2.2}$$

Suppose $\kappa_1, \kappa_2 \cdots \kappa_{m-k} \in \mathbb{C}$ are estimates of $m-k$ eigenvalues of $\mathcal{A}$ obtained from this process corresponding to a part of the spectrum we are not interested in. We can use these numbers as shifts to apply $m-k$ shifted QR steps to $H_m^{(i)}$ and get a Krylov decomposition

$$\mathcal{A}\tilde{U}_m^{(i)} = \tilde{U}_m^{(i)} \tilde{H}_m^{(i)} + h_{m+1,m}^{(i)} u_{m+1}^{(i)} (e_m^T Q^{(i)}), \tag{2.3}$$

where $Q^{(i)} = Q_1 Q_2 \cdots Q_{m-k}$ is the product of $m-k$ upper Hessenberg unitary matrices, $\tilde{U}_m^{(i)} = U_m^{(i)} Q^{(i)}$, $\tilde{H}_m^{(i)} = Q^{(i)*} H_m^{(i)} Q^{(i)}$ is upper Hessenberg, and $e_m^T Q^{(i)}$ is the last row of $Q^{(i)}$ with $k-1$ zero leading entries. For details, see [34] or [19, 36].

The restarted Arnoldi decomposition is then obtained from the first k columns of the above Krylov decomposition as follows

$$\begin{aligned} \mathcal{A}\tilde{U}_k^{(i)} &= \tilde{U}_k^{(i)} \tilde{H}_k^{(i)} + \tilde{h}_{k+1,k}^{(i)} \tilde{u}_{k+1}^{(i)} e_k^T + (h_{m+1,m}^{(i)} q_{m,k}^{(i)}) u_{m+1}^{(i)} e_k^T \\ \text{or} \quad \mathcal{A}U_k^{(i+1)} &= U_k^{(i+1)} H_k^{(i+1)} + h_{k+1,k}^{(i+1)} u_{k+1}^{(i+1)} e_k^T. \end{aligned} \tag{2.4}$$

Here $q_{m,k}^{(i)}$ is the (m,k) entry of $Q^{(i)}$, $U_k^{(i+1)} = \tilde{U}_k^{(i)}$, and $H_k^{(i+1)} = \tilde{H}_k^{(i)}$. Note that both $\tilde{u}_{k+1}^{(i)}$ and $u_{m+1}^{(i)}$ are orthogonal to $U_k^{(i+1)}$. Let $\hat{u}_k^{(i+1)} = \tilde{h}_{k+1,k}^{(i)} \tilde{u}_{k+1}^{(i)} + (h_{m+1,m}^{(i)} q_{m,k}^{(i)}) u_{m+1}^{(i)}$; then $h_{k+1,k}^{(i+1)} = \|\hat{u}_k^{(i+1)}\|$ and $u_k^{(i+1)} = (h_{k+1,k}^{(i+1)})^{-1} \hat{u}_k^{(i+1)}$. Clearly, no additional matrix-vector product involving $\mathcal{A}$ is used for the restart. For the restarted Arnoldi decomposition, it can be shown that $u_1^{(i+1)} = (\mathcal{A} - \kappa_1 I)(\mathcal{A} - \kappa_2 I) \cdots (\mathcal{A} - \kappa_{m-k} I) u_1^{(i)}$ up to a constant scaling factor. In other words, the eigenvector component corresponding to the unwanted spectrum in $u_1^{(i)}$ is filtered out by the filter polynomial.

An inexact implicitly restarted Arnoldi method is given as follows.

ALGORITHM 1. Inexact implicitly restarted Arnoldi (IRA) method.

Given a normalized $u_1^{(0)} \in \mathbb{C}^n$, $\epsilon > 0$ and $\tau > 0$. Let $j = 1$
For IRA cycles $i = 0, 1, 2, \ldots$

1. Compute $\mathcal{A}u_j^{(i)}$ by solving $Ay = Bu_j^{(i)}$ to a prescribed relative tolerance $\delta^{(i,j)}(\epsilon)$.
2. Expand the Arnoldi decomposition by orthogonalizing y against $u_1^{(i)}, \ldots, u_j^{(i)}$ and normalizing; we have $\mathcal{A}U_j^{(i)} + F_j^{(i)} = U_j^{(i)} H_j^{(i)} + h_{j+1,j}^{(i)} u_{j+1}^{(i)} e_j^T$, where the columns of $F_j^{(i)}$ (not computable) are the errors introduced at each Arnoldi step.
3. $j \leftarrow j + 1$. If $j = m$, make sure that $\|\mathcal{A}U_m^{(i)} - U_m^{(i)} H_m^{(i)} - h_{m+1,m} u_{m+1} e_m^T\| \lesssim \epsilon$. If this condition cannot be satisfied, claim that IRA fails; otherwise, invoke the implicit restart to get $\mathcal{A}U_k^{(i+1)} + F_k^{(i+1)} = U_k^{(i+1)} H_k^{(i+1)} + h_{k+1,k}^{(i+1)} u_{k+1}^{(i+1)} e_k^T$, and $j \leftarrow k + 1$. If $|h_{k+1,k}^{(i+1)}| \le \tau$, stop and output $U_k^{(i+1)}$; otherwise, continue.

End For

In this study, we choose the "exact shifts" strategy for the IRA method, which uses the unwanted eigenvalues of $H_m^{(i)}$ (Ritz values) as shifts for the implicit restart. This is the default choice in ARPACK and has proved successful in many applications. Some properties of the IRA method with the exact shifts strategy are given as follows.

PROPOSITION 2.1 (see Corollary 2.3, Chapter 5 of [36]). *Suppose $\mu_1, \ldots, \mu_m$ are eigenvalues of $H_m^{(i)}$. If the implicit QR steps are performed with shifts $\{\mu_{k+1}, \mu_{k+2}, \ldots, \mu_m\}$, then*

$$\tilde{H}_m^{(i)} = Q^{(i)*} H_m^{(i)} Q^{(i)} = \begin{bmatrix} \tilde{H}_k^{(i)} & \tilde{H}_m^{12(i)} \\ 0 & \tilde{H}_m^{22(i)} \end{bmatrix}, \tag{2.5}$$

where $\tilde{H}_m^{22(i)}$ is an upper triangular matrix with $\mu_{k+1}, \mu_{k+2}, \ldots, \mu_m$ on its diagonal.

The proposition shows that $\tilde{h}_{k+1,1}^{(i)} = 0$ if exact shifts are used. This observation immediately leads to the following result.

PROPOSITION 2.2. *Let the Schur decomposition of $H_m^{(i)}$ be $H_m^{(i)} = W_m^{(i)} T_m^{(i)} W_m^{(i)*}$, where $W_m^{(i)} = [W_m^{1(i)}, W_m^{2(i)}]$ is unitary, and*

$$T_m^{(i)} = \begin{bmatrix} T_m^{11(i)} & T_m^{12(i)} \\ 0 & T_m^{22(i)} \end{bmatrix} \tag{2.6}$$

with $\lambda(T_m^{11(i)}) = \{\mu_1, \mu_2, \ldots, \mu_k\}$, $\lambda(T_m^{22(i)}) = \{\mu_{k+1}, \mu_{k+2}, \ldots, \mu_m\}$, and $\lambda(T_m^{11(i)}) \cap \lambda(T_m^{22(i)}) = \emptyset$. Then

$$\|\mathcal{A}U_m^{(i)} W_m^{1(i)} - U_m^{(i)} W_m^{1(i)} T_m^{11(i)}\| = \|\mathcal{A}U_k^{(i+1)} - U_k^{(i+1)} H_k^{(i+1)}\| \tag{2.7}$$

and

$$\|h_{m+1,m}^{(i)} u_{m+1}^{(i)} e_m^T W_m^{1(i)}\| = \|h_{k+1,k}^{(i+1)} u_{k+1}^{(i+1)} e_k^T\|. \tag{2.8}$$

Proof. Let $Q^{(i)} = \left[Q^{1(i)}, Q^{2(i)}\right]$. From (2.5) and (2.6) we have $Q^{1(i)*} H_m^{(i)} Q^{1(i)} = \tilde{H}_k^{(i)}$ and $W_m^{1(i)*} H_m^{(i)} W^{1(i)} = T_m^{11(i)}$. Since $\lambda(\tilde{H}_k^{(i)}) = \lambda(T_m^{11(i)}) = \{\mu_1, \mu_2, \ldots, \mu_k\}$, there exists a $k \times k$ unitary matrix $V^{(i)}$ such that $V^{(i)*} \tilde{H}_k^{(i)} V^{(i)} = T_m^{11(i)}$ and $W_m^{1(i)} =$

$Q^{1(i)}V^{(i)}$. Note from (2.3) and (2.4) that $U_m^{(i)}Q^{1(i)} = U_k^{(i+1)}$. Therefore

$$\begin{aligned}(2.9)\qquad &\|\mathcal{A}U_m^{(i)}W_m^{1(i)} - U_m^{(i)}W_m^{1(i)}T_m^{11(i)}\| \\ &= \|\mathcal{A}U_m^{(i)}Q^{1(i)}V^{(i)} - U_m^{(i)}Q^{1(i)}V^{(i)}T_m^{11(i)}V^{(i)*}V^{(i)}\| \\ &= \|(\mathcal{A}U_k^{(i+1)} - U_k^{(i+1)}\tilde{H}_k^{(i)})V^{(i)}\| = \|\mathcal{A}U_k^{(i+1)} - U_k^{(i+1)}\tilde{H}_k^{(i)}\|.\end{aligned}$$

Since $\tilde{h}_{k+1,k}^{(i)} = 0$, we have $h_{k+1,k}^{(i+1)}u_{k+1}^{(i+1)}e_k^T = (h_{m+1,m}^{(i)}q_{m,k}^{(i)})u_{m+1}^{(i)}e_k^T$ from (2.4), and therefore

$$\begin{aligned}(2.10)\quad \|h_{m+1,m}^{(i+1)}u_{m+1}^{(i)}e_m^TW_m^{1(i)}\| &= \|h_{m+1,m}^{(i+1)}u_{m+1}^{(i)}e_m^TQ^{1(i)}V^{(i)}\| \\ &= \|(h_{m+1,m}^{(i+1)}q_{m,k}^{(i)})u_{m+1}^{(i)}e_k^T\| = \|h_{k+1,k}^{(i+1)}u_{k+1}^{(i+1)}e_k^T\|. \qquad \square\end{aligned}$$

These results are applicable to the standard (exact) IRA method as well as the inexact version of Algorithm 1. For the exact IRA method (where the matrix-vector products involving $\mathcal{A}$ are computed exactly), (2.8) can be derived from (2.7). For inexact IRA, however, the "true eigenvalue residuals" in (2.7) and the "estimated eigenvalue residuals" in (2.8) are different. Proposition 2.2 shows that the two types of eigenvalue residual norms are "restart-invariant" if exact shifts are used: both quantities at the end of the ith IRA cycle are the same as those at the beginning of the $(i+1)$th IRA cycle.

3. New strategies for solving linear systems in inexact IRA. To improve the efficiency for solving the linear systems arising in inexact eigenvalue algorithms, a special type of preconditioner with "tuning" is studied in [14, 15, 27, 17]. An existing preconditioner P is modified using a special low-rank update of P to produce a tuned preconditioner $\mathbb{P}$ that behaves like the system matrix A on a certain set of vectors X. It is shown in these papers that the inner iteration counts needed to solve the linear system preconditioned by $\mathbb{P}$ are substantially smaller than those required to solve the system preconditioned by P.

For example, consider inexact subspace iteration with $\mathcal{A} = A^{-1}$ used to detect a few smallest eigenvalues of A. In each outer iteration, we approximately solve the block linear system $AY^{(i)} = X^{(i)}$, where $X^{(i)}$ contains the current approximate Schur vectors (therefore $X^{(i)*}X^{(i)} = I$). It is shown in [27] that a decreasing sequence of tolerances for the block systems is necessary to guarantee the linear convergence of $X^{(i)}$ to the desired invariant subspace. As a result, the block-GMRES iteration counts required to solve $AP^{-1}\tilde{Y}^{(i)} = X^{(i)}$ (with $Y^{(i)} = P^{-1}\tilde{Y}^{(i)}$) increase gradually as the outer iteration progresses. To resolve this difficulty, P is replaced by the tuned preconditioner

$$(3.1)\qquad \mathbb{P}^{(i)} = P + (A - P)X^{(i)}X^{(i)*},$$

for which $\mathbb{P}^{(i)}X^{(i)} = AX^{(i)}$, or, equivalently, $A(\mathbb{P}^{(i)})^{-1}(AX^{(i)}) = AX^{(i)}$. That is, $AX^{(i)}$ spans an invariant subspace of the tuned preconditioned system matrix corresponding to eigenvalue 1. For $A(\mathbb{P}^{(i)})^{-1}\tilde{Y}^{(i)} = X^{(i)}$, the right-hand side $X^{(i)}$ spans an approximate invariant subspace of $A(\mathbb{P}^{(i)})^{-1}$, and the block-GMRES iteration counts needed for solving this preconditioned system do *not* increase with the outer iteration progress.

This idea of tuning is extended in [17] to an inexact IRA method for standard eigenvalue problems. Let m and k be the order of the Arnoldi decomposition, i.e.,

the number of columns in the Hessenberg matrix right before and after the implicit restart. Assume after the jth ($0 \le j \le m-k-1$) Arnoldi step in the ith IRA cycle an Arnoldi decomposition $\mathcal{A}U_{k+j}^{(i)} = U_{k+j}^{(i)} H_{k+j}^{(i)} + h_{k+j+1,k+j}^{(i)} u_{k+j+1}^{(i)} e_{k+j}^T$ is already computed, and $Ay = u_{k+j+1}^{(i)}$ needs to be solved in the $(j+1)$th Arnoldi step. In [17], the tuned preconditioning matrix is defined as $\mathbb{P}_{k+j+1}^{(i)} = P + (A-P)XX^*$, where $X = U_{k+j+1}^{(i)}$ contains the Arnoldi vectors in the ith IRA cycle. It is shown that the inner iteration counts required to solve $A(\mathbb{P}_{k+j+1}^{(i)})^{-1}\tilde{y} = u_{k+j+1}^{(i)}$ are smaller than those needed to solve $AP^{-1}\tilde{y} = u_{k+j+1}^{(i)}$, because $A(\mathbb{P}_{k+j+1}^{(i)})^{-1}$ has better eigenvalue clustering than AP^{-1}. This "clustering" effect of tuning is quite different from the original motivation of this strategy studied in [14, 15, 27]. In particular, $u_{k+j+1}^{(i)}$ is generally not a very good approximate eigenvector of $A(\mathbb{P}_{k+j+1}^{(i)})^{-1}$.

In this section, we propose and study a new tuning strategy for solving the linear systems of equations that arise in inexact IRA for generalized non-Hermitian eigenvalue problems. To study the new strategy under ideal conditions, we assume in this section that the linear system in each Arnoldi step is solved accurately (to machine precision). We also show how tuning can be used in a new two-phase algorithm to solve the linear systems in each Arnoldi step. We also discuss the use of subspace recycling with iterative solvers in Step 2 of Algorithm 2.

3.1. The new tuning strategy. The motivation for the tuning strategy is similar to that discussed in [14, 15, 27]: to make the right-hand side of the linear system associated with the spectral transformation an approximate eigenvector of the preconditioned system matrix, so that the inner iteration counts can be greatly reduced. Suppose we are in the ith IRA cycle and already have $\mathcal{A}U_{k+j}^{(i)} = U_{k+j}^{(i)} H_{k+j}^{(i)} + h_{k+j+1,k+j}^{(i)} u_{k+j+1}^{(i)} e_{k+j}^T$. Computing $\mathcal{A}u_{k+j+1}^{(i)}$ entails solving $Ay = Bu_{k+j+1}^{(i)}$. Recall that for a given X with orthonormal columns, the tuned preconditioner $\mathbb{P} = P + (A-P)XX^*$ satisfies $\mathbb{P}X = AX$, i.e., $A\mathbb{P}^{-1}(AX) = AX$. Tuning requires that X be chosen so that the right-hand side $Bu_{k+j+1}^{(i)}$ of the current linear system approximately lies in the subspace spanned by AX, an invariant subspace of $A\mathbb{P}^{-1}$.

Consider the choice

$$X = X_p^{(i,l)} = \left[\mathcal{A}U_m^{(i-l)}, \mathcal{A}U_{k+1:m}^{(i-l+1)}, \ldots, \mathcal{A}U_{k+1:m}^{(i-1)}, \mathcal{A}U_{k+1:k+j}^{(i)}\right], \tag{3.2}$$

where $U_{k+1:m}^{(r)}$ stands for the $(k+1)$th through the mth columns of $U_m^{(r)}$, and $p = m + (m-k)(l-1) + j$ is the number of vectors in $X_p^{(i,l)}$. We refer to $X_p^{(i,l)}$ as the set of "solution vectors," because its columns are solutions of the linear systems in previous Arnoldi steps. For example, the first vector in $X_p^{(i,l)}$ is $\mathcal{A}u_1^{(i-l)}$, the solution of $Ay = Bu_1^{(i-l)}$ in the first step of the $(i-l)$th IRA cycle. Note that this system does not need to be solved for $i > l$ due to the implicit restart.[1]

In the following derivation, we use a calligraphic letter to stand for a subspace spanned by some set of *column vectors* denoted by the same letter in Roman fonts. For instance, $\mathcal{U}_{k+j}^{(i)} = \text{span}\{U_{k+j}^{(i)}\}$. Let $U_p^{(i,l)} = [U_m^{(i-l)}, U_{k+1:m}^{(i-l+1)}, \ldots, U_{k+1:k+j}^{(i)}]$ and let $\mathcal{U}_p^{(i,l)} = \text{span}\{U_p^{(i,l)}\}$, $\mathcal{X}_p^{(i,l)} = \text{span}\{X_p^{(i,l)}\}$, $A\mathcal{X}_p^{(i,l)} = B\mathcal{U}_p^{(i,l)} = \text{span}\{BU_m^{(i-l)}, BU_{k+1:m}^{(i-l+1)}, \ldots, BU_{k+1:k+j}^{(i)}\}$. To study the relation between $Bu_{k+j+1}^{(i)}$ and $A\mathcal{X}_p^{(i,l)}$, we

[1] Implicit restart generates a new Arnoldi decomposition of size k, and thus $Ay = Bu_j^{(i-l)}$ does not need to be solved for $i > l$ (any restarted cycle) and $1 \le j \le k$.

begin with the following lemma, which shows that the range of $U_p^{(i,l)}$ is a Krylov subspace.

LEMMA 3.1. *Suppose that IRA does not break down. Then* $\mathcal{U}_p^{(i,l)} = \mathcal{K}_p(\mathcal{A}, u_1^{(i-l)})$.

Proof. First, $\mathcal{U}_{m+1}^{(i-l)} = \mathcal{K}_{m+1}(\mathcal{A}, u_1^{(i-l)})$. Since $u_{k+1}^{(i-l+1)}$ is a linear combination of $u_{m+1}^{(i-l)}$ and $\tilde{u}_{k+1}^{(i-l)} \in \mathcal{U}_m^{(i-l)}$ (see (2.4)), $\mathrm{span}\{U_m^{(i-l)}, u_{k+1}^{(i-l+1)}\} = \mathcal{K}_{m+1}(\mathcal{A}, u_1^{(i-l)})$ holds. As we have orthogonalized $\mathcal{A}u_{k+1}^{(i-l+1)}$ against $\mathcal{U}_{k+1}^{(i-l+1)} \subset \mathrm{span}\{U_m^{(i-l)}, u_{k+1}^{(i-l+1)}\}$ (note that $\mathcal{U}_k^{(i-l+1)} \subset \mathcal{U}_m^{(i-l)}$; see (2.4)) to get $u_{k+2}^{(i-l+1)}$, $\mathrm{span}\{U_m^{(i-l)}, u_{k+1}^{(i-l+1)}, u_{k+2}^{(i-l+1)}\} = \mathcal{K}_{m+2}(\mathcal{A}, u_1^{(i-l)})$ follows. Similar reasoning holds for all of the following Arnoldi vectors if IRA does not break down, and the theorem is established. ∎

The angle between a vector v and a subspace $\mathcal{U}$ (denoted as $\angle(v,\mathcal{U})$) is defined as the angle between v and the orthogonal projection of v onto $\mathcal{U}$. Obviously, $v \in \mathcal{U}$ if and only if $\angle(v,\mathcal{U}) = 0$. Therefore, $Bu_{k+j+1}^{(i)}$ approximately lies in $A\mathcal{X}_p^{(i,l)} = B\mathcal{U}_p^{(i,l)}$ if and only if $\angle(Bu_{k+j+1}^{(i)}, B\mathcal{U}_p^{(i,l)})$ is small, and this condition holds if $\varphi_p^{(i)} = \angle(u_{k+j+1}^{(i)}, \mathcal{U}_p^{(i,l)})$ is small and if B does not significantly distort this angle. The following theorem gives a sufficient condition for $\angle(Bu_{k+j+1}^{(i)}, B\mathcal{U}_p^{(i,l)})$ to be small.

THEOREM 3.2. *Let* $B = U_B\Sigma_B V_B^*$ *be the singular value decomposition of* B, *and* $u_{k+j+1}^{(i)} = u_p c_p^{(i,l)} + u_p^\perp s_p^{(i,l)}$, *where* $u_p = V_B f_p \in \mathcal{U}_p^{(i,l)}$ *and* $u_p^\perp = V_B f_p^\perp \perp \mathcal{U}_p^{(i,l)}$ *are unit vectors, and* $s_p^{(i,l)}$ *and* $c_p^{(i,l)}$ *are sine and cosine of* $\angle(u_{k+j+1}^{(i)}, \mathcal{U}_p^{(i,l)})$. *Assume that* $s_p^{(i,l)}$ *is small enough such that* $\|Bu_p^\perp s_p^{(i,l)}\| < \|Bu_p c_p^{(i,l)}\|$. *Then*

$$\sin\angle(Bu_{k+j+1}^{(i)}, B\mathcal{U}_p^{(i,l)}) \le \frac{\|\Sigma_B f_p^\perp\|}{\|\Sigma_B f_p\|} \tan\angle(u_{k+j+1}^{(i)}, \mathcal{U}_p^{(i,l)}). \tag{3.3}$$

Proof. Given the orthogonal decomposition of $u_{k+j+1}^{(i)}$, we have $Bu_{k+j+1}^{(i)} = Bu_p c_p^{(i,l)} + Bu_p^\perp s_p^{(i,l)} = U_B\Sigma_B f_p c_p^{(i,l)} + U_B\Sigma_B f_p^\perp s_p^{(i,l)}$. Consider a sphere centered at the terminal point of the vector $Bu_p c_p^{(i,l)}$, with radius $\|Bu_p^\perp s_p^{(i,l)}\|$. (Note that the origin is outside of the sphere because it is assumed that $\|Bu_p^\perp s_p^{(i,l)}\| < \|Bu_p c_p^{(i,l)}\|$.) The terminal point of $Bu_{k+j+1}^{(i)}$ must be on this sphere. It follows that $\sin\angle(Bu_{k+j+1}^{(i)}, Bu_p) \le \frac{\|Bu_p^\perp s_p^{(i,l)}\|}{\|Bu_p c_p^{(i,l)}\|}$, and the equality holds if and only if $Bu_p^\perp \perp Bu_{k+j+1}^{(i)}$ (i.e., $Bu_{k+j+1}^{(i)}$ is tangent to this sphere). Therefore, we have

$$\begin{aligned} \sin\angle(Bu_{k+j+1}^{(i)}, B\mathcal{U}_p^{(i,l)}) &\le \sin\angle(Bu_{k+j+1}^{(i)}, Bu_p) \\ &\le \frac{\|Bu_p^\perp s_p^{(i,l)}\|}{\|Bu_p c_p^{(i,l)}\|} = \frac{\|\Sigma_B f_p^\perp\|}{\|\Sigma_B f_p\|} \tan\angle(u_{k+j+1}^{(i)}, \mathcal{U}_p^{(i,l)}). \end{aligned} \tag{3.4}$$

This completes the proof. ∎

We see from Theorem 3.2 that if $\angle(u_{k+j+1}^{(i)}, \mathcal{U}_p^{(i,l)})$ is small, then a sufficient condition for $\angle(Bu_{k+j+1}^{(i)}, B\mathcal{U}_p^{(i,l)})$ to be small is that u_p (the normalized orthogonal projection of $u_{k+j+1}^{(i)}$ onto $\mathcal{U}_p^{(i,l)}$) and $u_p^\perp$ (the normalized $u_{k+j+1}^{(i)} - u_p c^{(i,l)}$) can each be written as linear combinations of the right singular vectors of B with coefficients f_p and $f_p^\perp$, respectively, and the corresponding weighted singular value $\|\Sigma_B f_p^\perp\|$ is a small multiple of $\|\Sigma_B f_p\|$. In particular, $\angle(Bu_{k+j+1}^{(i)}, B\mathcal{U}_p^{(i,l)})$ is small if $\kappa(B)$ is small.

Next, we establish a relation between successive angles $\varphi_p^{(i)}$ and $\varphi_{p-1}^{(i)} = \angle(u_{k+j}^{(i)}, \mathcal{U}_{p-1}^{(i,l)})$, $(p = m + (m-k)(l-1) + j,\ 1 \le j \le m-k)$.

THEOREM 3.3. *Let $u_{k+j}^{(i)} = u_{p-1}\cos\varphi_{p-1}^{(i)} + u_{p-1}^{\perp}\sin\varphi_{p-1}^{(i)}$, where $u_{p-1} \in \mathcal{U}_{p-1}^{(i,l)}$ and $u_{p-1}^{\perp} \perp \mathcal{U}_{p-1}^{(i,l)}$ are unit vectors. Let the orthogonal projection of $\|\mathcal{A}u_{p-1}^{\perp}\|^{-1}\mathcal{A}u_{k+j}^{(i)}$ onto $\mathcal{U}_p^{(i,l)}$ be $w_p\eta_p$, where $w_p \in \mathcal{U}_p^{(i,l)}$ is a unit vector, and let $\alpha_p = \angle(\mathcal{A}u_{p-1}^{\perp}, \mathcal{U}_p^{(i,l)})$ and $\beta_p = \angle(w_p, \mathcal{U}_{k+j}^{(i)})$. Then*

$$\tan\varphi_p^{(i)} = \nu_p \sin\varphi_{p-1}^{(i)}, \quad \text{where } \nu_p = \frac{\sin\alpha_p}{\eta_p \sin\beta_p}. \tag{3.5}$$

(Note that α_p, β_p, η_p, and ν_p all depend on the IRA cycle number i. To simplify the notation, we omit the superscripts for these scalars.)

Proof. Let $\rho = \frac{\|\mathcal{A}u_{p-1}\|}{\|\mathcal{A}u_{p-1}^{\perp}\|}$. Since $\mathcal{U}_p^{(i,l)} = \mathcal{K}_p(\mathcal{A}, u_1^{(i-l)})$, we have

$$\begin{aligned}
\mathcal{A}u_{k+j}^{(i)} &= \mathcal{A}u_{p-1}\cos\varphi_{p-1}^{(i)} + \mathcal{A}u_{p-1}^{\perp}\sin\varphi_{p-1}^{(i)} \\
&= \rho\|\mathcal{A}u_{p-1}^{\perp}\| w_{p1}\cos\varphi_{p-1}^{(i)} + \|\mathcal{A}u_{p-1}^{\perp}\|(w_{p2}\cos\alpha_p + w_p^{\perp}\sin\alpha_p)\sin\varphi_{p-1}^{(i)} \\
&= \|\mathcal{A}u_{p-1}^{\perp}\|(w_{p1}\rho\cos\varphi_{p-1}^{(i)} + w_{p2}\cos\alpha_p\sin\varphi_{p-1}^{(i)} + w_p^{\perp}\sin\alpha_p\sin\varphi_{p-1}^{(i)}) \\
&= \|\mathcal{A}u_{p-1}^{\perp}\|(w_p\eta_p + w_p^{\perp}\sin\alpha_p\sin\varphi_{p-1}^{(i)}),
\end{aligned} \tag{3.6}$$

where w_{p1},w_{p2},$w_p \in \mathcal{U}_p^{(i,l)}$, and $w_p^{\perp} \perp \mathcal{U}_p^{(i,l)}$ are all unit vectors, and $w_p\eta_p = w_{p1}\rho\cos\varphi_{p-1}^{(i)} + \ w_{p2}\cos\alpha_p\sin\varphi_{p-1}^{(i)}$ is the orthogonal projection of $\frac{\mathcal{A}u_{k+j}^{(i+l)}}{\|\mathcal{A}u_{p-1}^{\perp}\|}$ onto $\mathcal{U}_p^{(i,l)}$. It follows immediately that $\tan\angle(\mathcal{A}u_{k+j}^{(i)}, \mathcal{U}_p^{(i,l)}) = \frac{\sin\alpha_p\sin\varphi_{p-1}^{(i)}}{\eta_p}$.

We then orthogonalize $\mathcal{A}u_{k+j}^{(i)}$ against $\mathcal{U}_{k+j}^{(i)} \subset \mathcal{U}_p^{(i,l)}$ to get $u_{k+j+1}^{(i)}$. Let $\mathcal{U}_{k+j}^{(i)\perp}$ be the orthogonal complement of $\mathcal{U}_{k+j}^{(i)}$ in $\mathcal{U}_p^{(i,l)}$. Then $w_p = w_{p3}\cos\beta_p + w_{p4}\sin\beta_p$, where $w_{p3} \in \mathcal{U}_{k+j}^{(i)}$ and $w_{p4} \in \mathcal{U}_{k+j}^{(i)\perp}$ are unit vectors, and $\beta_p = \angle(w_p, \mathcal{U}_{k+j}^{(i)})$. Orthogonalizing $\mathcal{A}u_{k+j}^{(i)}$ against $\mathcal{U}_{k+j}^{(i)}$ removes the w_{p3} component from w_p, so that $u_{k+j+1}^{(i)}$ equals $w = w_{p4}\eta_p\sin\beta_p + w_p^{\perp}\sin\alpha_p\sin\varphi_{p-1}^{(i)}$ up to a constant scaling factor. It follows that $\tan\angle(u_{k+j+1}^{(i)}, \mathcal{U}_p^{(i,l)}) = \frac{\sin\alpha_p\sin\varphi_{p-1}^{(i)}}{\eta_p\sin\beta_p}$, and (3.5) is established. ∎

Remark 3.1. In Theorem 3.3 we are interested in the nontrivial case where $l > 0$. If $l = 0$, then $p = k + j$, and $\mathcal{U}_p^{(i,0)} = \mathcal{U}_{k+j}^{(i)}$. Therefore $\beta_p = \angle(w_p, \mathcal{U}_{k+j}^{(i)}) = 0$ (because $w_p \in \mathcal{U}_p^{(i,0)}$ by definition), ν_p is infinity, and $\varphi_p^{(i)} = \pi/2$. This is consistent with the fact that Arnoldi vectors in the same IRA cycle are orthogonal. In addition, since exact shifts are used for the implicit restart, we have $u_{k+1}^{(i)} = u_{m+1}^{(i-1)}$; see (2.4) and Proposition 2.1. Therefore, Theorem 3.3 also holds for $j = 0$, with $u_{k+j}^{(i)}$ replaced by $u_m^{(i-1)}$.

Given the starting vector $\mathcal{A}u_1^{(i-l)}$ of $X_p^{(i,l)}$, Theorem 3.3 shows that if ν_p is bounded above by a constant smaller than 1, then $\varphi_p^{(i)}$ decreases at least linearly with p for $p > m$. In practice, although we have no upper bounds for ν_p, we have consistently found in our experiments that $\nu_p < 1$ at a majority of Arnoldi steps, and $\varphi_p^{(i)}$ decreases linearly with p on the whole. In addition, $\sin\angle(Bu_{k+j+1}^{(i)}, B\mathcal{U}_p^{(i,l)})$ is in

general a moderate multiple of $\sin\angle(u_{k+j+1}^{(i)}, \mathcal{U}_p^{(i,l)})$, and therefore it is also small for large p. We will give a numerical example in section 5 to demonstrate these observations. Small $\angle(Bu_{k+j+1}^{(i)}, B\mathcal{U}_p^{(i,l)})$ means that $Bu_{k+j+1}^{(i)}$ is an approximate eigenvector of $A\mathbb{P}^{-1}$ (where the tuned preconditioner $\mathbb{P}$ is constructed using $X_p^{(i,l)}$), because it approximately lies in $A\mathcal{X}_p^{(i,l)} = B\mathcal{U}_p^{(i,l)}$, an invariant subspace of $A\mathbb{P}^{-1}$. In the following subsection, we show how this observation can be used in a new two-phase algorithm for solving $Ay = Bu_{k+j+1}^{(i)}$.

3.2. A two-phase strategy to solve the linear systems in Arnoldi steps. With the new tuning discussed in subsection 3.1, we now propose a new two-phase algorithm for efficiently solving $Ay = Bu_{k+j+1}^{(i)}$ in step 1 of Algorithm 1.

ALGORITHM 2. Two-phase strategy for solving $Ay = Bu_{k+j+1}^{(i)}$.

1. Construct the tuned preconditioner $\mathbb{P}$ using (3.2) and find the minimum residual solution $y_1 = \mathbb{P}^{-1}\tilde{y}_1 = \gamma\mathbb{P}^{-1}Bu_{k+j+1}^{(i)}$, where $\gamma = \arg\min_\gamma \|Bu_{k+j+1}^{(i)} - \gamma A\mathbb{P}^{-1}Bu_{k+j+1}^{(i)}\|$.
2. For ϵ as in Algorithm 1, choose either a fixed tolerance $\delta = \delta_f(\epsilon)$, or a relaxed tolerance $\delta = \delta_r(\epsilon)$ by some means. Solve the *correction equation* $Az = Bu_{k+j+1}^{(i)} - Ay_1$ with *any* appropriate preconditioned iterative solver to get an approximate correction z_q, such that the corrected iterate $y_{q+1} = y_1 + z_q$ satisfies $\frac{\|Bu_{k+j+1}^{(i)} - Ay_{q+1}\|}{\|Bu_{k+j+1}^{(i)}\|} \le \delta$, or, equivalently, the correction z_q satisfies $\frac{\|(Bu_{k+j+1}^{(i)} - Ay_1) - Az_q\|}{\|Bu_{k+j+1}^{(i)} - Ay_1\|} \le \frac{\delta\|Bu_{k+j+1}^{(i)}\|}{\|Bu_{k+j+1}^{(i)} - Ay_1\|}$.

In particular, tuning is used only in phase I to obtain a good approximate solution y_1 for $Ay = Bu_{k+j+1}^{(i)}$. In fact, we have found that by reformulating the solution algorithm in this way, tuning actually does not improve performance for computing the correction, so that the extra expense of tuning can be avoided with no penalty. Therefore, we can work with a fixed preconditioned system matrix for the correction equation in all Arnoldi steps.

Let $u_{k+j+1}^{(i)} = u_p c_p^{(i,l)} + u_p^\perp s_p^{(i,l)}$, where $u_p \in \mathcal{U}_p^{(i,l)}$ and $u_p^\perp \perp \mathcal{U}_p^{(i,l)}$ are unit vectors, and $s_p^{(i,l)}$ and $c_p^{(i,l)}$ are the sine and cosine of $\angle(u_{k+j+1}^{(i)}, \mathcal{U}_p^{(i,l)})$. We have shown by Theorem 3.3 that $s_p^{(i,l)}$ can be small for large p. The analysis of Algorithm 2 is given in the following major theorem.

THEOREM 3.4. *Suppose Algorithm* 2 *is used to solve* $Ay = Bu_{k+j+1}^{(i)}$. *Then Phase* I *of Algorithm* 2 *gives* $y_1 = \mathcal{A}u_p c_p^{(i,l)} + O(s_p^{(i,l)})$ *(up to a constant scaling factor) and the corresponding relative residual norm* $\frac{\|Bu_{k+j+1}^{(i)} - Ay_1\|}{\|Bu_{k+j+1}^{(i)}\|} = O(s_p^{(i,l)})$. *Consequently, the stopping criterion of Algorithm* 2 *is satisfied if and only if the relative residual of the correction equation* $\frac{\|(Bu_{k+j+1}^{(i)} - Ay_1) - Az_q\|}{\|Bu_{k+j+1}^{(i)} - Ay_1\|} \le \frac{\delta\|Bu_{k+j+1}^{(i)}\|}{\|Bu_{k+j+1}^{(i)} - Ay_1\|} = \frac{\delta}{O(s_p^{(i,l)})}$.

Proof. It is shown in section 3.1 that if the preconditioning matrix $\mathbb{P}_p^{(i,l)}$ is constructed using $X_p^{(i,l)}$, then $A(\mathbb{P}_p^{(i,l)})^{-1}(AX_p^{(i,l)}) = AX_p^{(i,l)}$. That is, $A\mathcal{X}_p^{(i,l)} = B\mathcal{U}_p^{(i,l)}$ is an invariant subspace of dimension p of $A(\mathbb{P}_p^{(i,l)})^{-1}$ corresponding to eigenvalue 1. It follows that for $u_p \in \mathcal{U}_p^{(i,l)}$, $(\mathbb{P}_p^{(i,l)})^{-1}(Bu_p) = A^{-1}(Bu_p)$.

The approximate solution to $A(\mathbb{P}_p^{(i,l)})^{-1}\tilde{y} = Bu_{k+j+1}^{(i)}$ after Phase I of Algorithm 2 is

$$\begin{aligned} y_1 &= (\mathbb{P}_p^{(i,l)})^{-1}\tilde{y}_1 = (\mathbb{P}_p^{(i,l)})^{-1}(\gamma Bu_{k+j+1}^{(i)}) = \gamma(\mathbb{P}_p^{(i,l)})^{-1}(Bu_p c_p^{(i,l)} + Bu_p^{\perp} s_p^{(i,l)}) \\ &= \gamma\left(A^{-1}Bu_p c_p^{(i,l)} + (\mathbb{P}_p^{(i,l)})^{-1}Bu_p^{\perp} s_p^{(i,l)}\right) = \gamma(\mathcal{A}u_p c_p^{(i,l)} + \delta_p^{(i,l)}), \end{aligned} \tag{3.7}$$

where $\|\delta_p^{(i,l)}\| = s_p^{(i,l)}\|(\mathbb{P}_p^{(i,l)})^{-1}Bu_p^{\perp}\| = O(s_p^{(i,l)})$.

Now consider the residual norm after Phase I of Algorithm 2:

$$\begin{aligned} &\|Bu_{k+j+1}^{(i)} - Ay_1\| \\ &= \min_{\gamma} \|Bu_{k+j+1}^{(i)} - \gamma A(\mathbb{P}_p^{(i,l)})^{-1}(Bu_{k+j+1}^{(i)})\| \\ &\le \|Bu_{k+j+1}^{(i)} - A(\mathbb{P}_p^{(i,l)})^{-1}(Bu_{k+j+1}^{(i)})\| \\ &= \|Bu_{k+j+1}^{(i)} - A(\mathbb{P}_p^{(i,l)})^{-1}(Bu_p c_p^{(i,l)} + Bu_p^{\perp} s_p^{(i,l)})\| \\ &= \|Bu_p c_p^{(i,l)} + Bu_p^{\perp} s_p^{(i,l)} - A(A^{-1}Bu_p)c_p^{(i,l)} - A(\mathbb{P}_p^{(i,l)})^{-1}Bu_p^{\perp} s_p^{(i,l)})\| \\ &= s_p^{(i,l)}\|(A(\mathbb{P}_p^{(i,l)})^{-1} - I)Bu_p^{\perp}\|. \end{aligned} \tag{3.8}$$

Therefore the relative residual norm is $s_p^{(i,l)}\frac{\|(A(\mathbb{P}_p^{(i,l)})^{-1}-I)Bu_p^{\perp}\|}{\|Bu_{k+j+1}^{(i)}\|} = O(s_p^{(i,l)})$.

Finally, Phase II of Algorithm 2 requires that

$$\frac{\|Bu_{k+j+1}^{(i)} - Ay_{q+1}\|}{\|Bu_{k+j+1}^{(i)}\|} = \frac{\|Bu_{k+j+1}^{(i)} - Ay_{q+1}\|}{\|Bu_{k+j+1}^{(i)} - Ay_1\|}\frac{\|Bu_{k+j+1}^{(i)} - Ay_1\|}{\|Bu_{k+j+1}^{(i)}\|} \le \delta, \tag{3.9}$$

which is satisfied if and only if the relative residual of the correction equation

$$\begin{aligned} \frac{\|(Bu_{k+j+1}^{(i)} - Ay_1) - Az_q\|}{\|Bu_{k+j+1}^{(i)} - Ay_1\|} &= \frac{\|Bu_{k+j+1}^{(i)} - Ay_{q+1}\|}{\|Bu_{k+j+1}^{(i)} - Ay_1\|} \\ &\le \frac{\delta\|Bu_{k+j+1}^{(i)}\|}{\|Bu_{k+j+1}^{(i)} - Ay_1\|} = \frac{\delta}{O(s_p^{(i,l)})}. \end{aligned} \tag{3.10}$$

The proof is thus concluded. ∎

Theorem 3.4 shows that y_1 obtained in Phase I of Algorithm 2 equals $\mathcal{A}u_p c_p^{(i,l)}$ plus a small quantity proportional to $s_p^{(i,l)}$. As $u_p \in \mathcal{U}_p^{(i,l)}$, $\mathcal{A}u_p \in \mathcal{A}\mathcal{U}_p^{(i,l)} = \mathcal{X}_p^{(i,l)}$; see (3.2). Recall that $X_p^{(i,l)}$ consists of the "solution vectors" of the linear systems in previous Arnoldi steps. Therefore, by constructing tuning as in section 3.1 and using it properly, we get a good approximate solution y_1 which is roughly a linear combination of those solution vectors. The reason for the success of this approach is that $\angle(Bu_{k+j+1}^{(i)}, B\mathcal{U}_p^{(i,l)})$ is small, i.e., $Bu_{k+j+1}^{(i)}$ is roughly a linear combination of the right-hand sides of the previously solved systems. This perspective is quite different from the motivation of tuning in previous literature [14, 15, 17, 27].

We see that a good approximate solution y_1 can be computed inexpensively in Phase I of Algorithm 2 by tuning, so that we always have $\frac{\|Bu_{k+j+1}^{(i)} - Ay_1\|}{\|Bu_{k+j+1}^{(i)}\|} = O(s_p^{(i,l)}) \ll 1$ in practice. In fact, a valid y_1 can also be obtained in other ways, in particular, by solving a least squares problem

$$\min_f \|Bu_{k+j+1}^{(i)} - AX_p^{(i,l)}f\|, \tag{3.11}$$

which can be easily done using the QR decomposition of $AX_p^{(i,l)} = BU_p^{(i,l)}$ (recall the definition of $X_p^{(i,l)}$ in (3.2)). Given that $u_{k+j+1}^{(i)} = u_p c_p^{(i,l)} + u_p^{\perp} s_p^{(i,l)}$, where $u_p \in \mathcal{U}_p^{(i,l)}$ and $u_p^{\perp} \perp \mathcal{U}_p^{(i,l)}$, we have

$$\begin{aligned} &\min_{f\in\mathbb{C}^p} \|Bu_{k+j+1}^{(i)} - AX_p^{(i,l)} f\| \\ &\quad = \min_{f\in\mathbb{C}^p} \|Bu_{k+j+1}^{(i)} - BU_p^{(i,l)} f\| \\ &\quad \le \|B(u_p c_p^{(i,l)} + u_p^{\perp} s_p^{(i,l)}) - Bu_p c_p^{(i,l)}\| = s_p^{(i,l)} \|Bu_p^{\perp}\|. \end{aligned} \tag{3.12}$$

Therefore, with $y_1 = X_p^{(i,l)} f$, we have $\frac{\|Bu_{k+j+1}^{(i)} - Ay_1\|}{\|Bu_{k+j+1}^{(i)}\|} = s_p^{(i,l)} \frac{\|Bu_p^{\perp}\|}{\|Bu_{k+j+1}^{(i)}\|} = O(s_p^{(i,l)})$. The Phase I computation in Algorithm 2 is somewhat cheaper for the least squares approach than the one-step tuned preconditioned GMRES, but the former method required slightly more iterations in Phase II for our test problems, and the total inner iteration counts are about the same for the two approaches. In the following, for the sake of brevity, we study only the two-phase strategy where tuning is applied in Phase I.

Due to the large reduction of the linear residual norm in Phase I, the stopping criterion in Algorithm 2, $\frac{\|Bu_{k+j+1}^{(i)} - Ay_{q+1}\|}{\|Bu_{k+j+1}^{(i)}\|} \le \delta$, is satisfied if and only if the relative residual of the correction equation $\frac{\|(Bu_{k+j+1}^{(i)} - Ay_1) - Az_q\|}{\|Bu_{k+j+1}^{(i)} - Ay_1\|}$ is bounded by the much less stringent relative tolerance $\frac{\delta \|Bu_{k+j+1}^{(i)}\|}{\|Bu_{k+j+1}^{(i)} - Ay_1\|} = \frac{\delta}{O(s_p^{(i,l)})} \gg \delta$. This larger relative tolerance implies that the inner iterations required for solving the correction equation can be considerably smaller than those needed to solve the original equation directly.

Remark 3.2. The two-phase algorithm has some similarities to other deflation methods that aim to improve the convergence of linear solves; see [8, 18, 39] and the references therein. For example, Algorithm 2 is very similar in procedure to the Init-CG algorithm [18]. They both build a subspace, remove the components belonging to that space from the right-hand side, and then solve a correction equation. The difference lies in the motivation and use of the subspaces. The two-phase algorithm here builds a subspace using the solution of the inner linear systems from previous Arnoldi steps. This space bears no obvious connection to the spectral properties of the coefficient matrix of the linear system. Because the angle between the right-hand side and this subspace is small, we obtain a good approximate solution and can solve the correction equation with considerably lower accuracy. In contrast, Init-CG forms a subspace spanned by explicitly computed eigenvector approximations of the coefficient matrix corresponding to extremal eigenvalues, and then deflates these components from the right-hand side. This enables accelerated convergence of CG.

3.3. Subspace recycling for the correction equation. Phase II of Algorithm 2 can be improved using linear solvers with subspace recycling. This methodology has proved efficient for solving a long sequence of *slowly changing* linear systems. When the iterative solution of one linear system is done, a small set of vectors from the current subspace for the candidate solutions is selected and "recycled," i.e., used for the solution of the next system in the sequence. Subspace recycling usually reduces the cost of solving subsequent linear systems, because the iterative solver does not have to build the candidate solution subspace from scratch. A popular solver of this type is the generalized conjugate residual method with implicit inner orthogonalization and

deflated restarting (GCRO-DR) [26] developed using ideas of special truncation [7] and restarting [25] for solving a single linear system.

Reference [26] makes a general assumption that the preconditioned system matrix changes from one linear system to the next, and thus 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. For the sequence of correction equations in Algorithm 2, fortunately, this transformation can be avoided, because the preconditioned system matrix without tuning is the same for the correction equation in all Arnoldi steps.

When GMRES is applied to a system whose coefficient matrix has a small number of eigenvalues of very small magnitude, convergence is often very slow in the initial steps of the iteration, until the asymptotic performance of GMRES is obtained. We refer to this initial stage of slow convergence as a period of *latency*; see [12] for discussion and analyis of this phenomenon. For recyling, it is suggested in [26] that the harmonic Ritz vectors corresponding to smallest harmonic Ritz values can be chosen to span the recycled subspaces. These vectors are approximate eigenvectors of the preconditioned system matrix corresponding to smallest eigenvalues. If the harmonic Ritz vectors are good approximate eigenvectors, this strategy tends to reduce the duration of this initial latency in convergence.

Our subspace recycling also uses dominant Ritz vectors, as suggested in [26]. In section 5, we present experimental results to show that these vectors are effective for subspace recycling if the use of harmonic Ritz vectors fails to reduce the inner iteration counts.

4. A refined analysis of allowable errors in Arnoldi steps. It was observed empirically in [4] that for the unrestarted Arnoldi method, the matrix-vector products involving $\mathcal{A}$ must be computed with high accuracy in the initial Arnoldi steps, but the accuracy can be relaxed as the iteration proceeds. A similar observation was also given in [20] for an inexact Lanczos method. An analysis based on matrix perturbation theory in [28] shows that the allowable errors of the matrix-vector products need only be inversely proportional to the eigenvalue residual norm of the current desired approximate invariant subspace for the quality of the approximate invariant subspace generated by the inexact Arnoldi method to be good. This *relaxation strategy* is extended in [17] to the inexact IRA method, where a practical estimate of the allowable errors of the inner solves at each Arnoldi step is proposed. Ideally, accurately estimated allowable errors can help reduce the inner iteration counts to the best extent possible without compromising the performance of the eigenvalue solvers. In this section, we give a refined analysis and an alternative estimate of allowable errors of the inner solves.

Suppose the matrix-vector product involving $\mathcal{A} = A^{-1}B$ is applied inexactly for m Arnoldi steps, with an error $f_j = y - A^{-1}Bu_j$ $(1 \le j \le m)$ introduced in the linear solve of $Ay = Bu_j$. Thus we have the following inexact Arnoldi decomposition:

$$\mathcal{A}U_m + F_m = (\mathcal{A} + F_m U_m^*)U_m = U_m H_m + h_{m+1,m} u_{m+1} e_m^T, \tag{4.1}$$

where U_m spans a Krylov subspace of the perturbed matrix $\mathcal{A} + F_m U_m^*$. Let the Schur decomposition of H_m be

$$H_m = W_m T_m W_m^*, \quad \text{with} \quad W_m = \begin{bmatrix} W_m^{11} & W_m^{12} \\ W_m^{21} & W_m^{22} \end{bmatrix} \quad \text{and} \quad T_m = \begin{bmatrix} T_m^{11} & T_m^{12} \\ 0 & T_m^{22} \end{bmatrix}, \tag{4.2}$$

where $T_m^{11} \in \mathbb{C}^{k\times k}$, $T_m^{22} \in \mathbb{C}^{p\times p}$, and $\lambda(T_m^{11})$ are the wanted Ritz values and $\lambda(T_m^{22})$ are

the unwanted ones. Then we use the Rayleigh–Ritz method (section 4.1, Chapter 4 of [36]) to extract the desired approximate invariant subspace $U_m W_m^1$, where $W_m^1 = \begin{bmatrix} W_m^{11} \\ W_m^{21} \end{bmatrix}$ contains the wanted Ritz vectors. From (4.1) and (4.2), the corresponding eigenvalue residual is

$$\begin{aligned} \mathcal{A}U_m W_m^1 - U_m H_m W_m^1 &= \mathcal{A}U_m W_m^1 - U_m W_m^1 T_m^{11} \\ &= h_{m+1,m} u_{m+1} e_m^T W_m^1 - F_m W_m^1, \end{aligned} \tag{4.3}$$

from which follows

$$\|(\mathcal{A}U_m W_m^1 - U_m W_m^1 T_m^{11}) - h_{m+1,m} u_{m+1} e_m^T W_m^1\| = \|F_m W_m^1\|. \tag{4.4}$$

Here, as introduced in section 2, $\mathcal{A}U_m W_m^1 - U_m W_m^1 T_m^{11}$ is the true eigenvalue residual, and $R_m = h_{m+1,m} u_{m+1} e_m^T W_m^1$ is the estimated residual (referred to as the "computed residual" in [28, 17]). The difference between the two residuals depends on $\|F_m W_m^1\|$. For the inexact Arnoldi method, we want to keep the quality of $U_m W_m^1$ under control in spite of the presence of the error matrix F_m. To achieve this goal, we need to control $\|F_m W_m^1\|$ so that the desired approximate invariant subspace $U_m W_m^1$ contained in U_m is not obviously contaminated by F_m; i.e., the true residual is still reasonably close to the estimated residual.

To see why the allowable errors at some Arnoldi steps can be relaxed, note that

$$\|F_m W_m^1\| \le \|F_k W_m^{11}\| + \|F_{k+1:m} W_m^{21}\| \le \|F_k\| + \|F_{k+1:m}\| \|W_m^{21}\|. \tag{4.5}$$

Therefore, for a given k-step inexact Arnoldi decomposition with a small enough $\|F_k\|$, $\|F_{k+1:m}\|$ does not have to be very small as long as $\|W_m^{21}\|$; i.e., the magnitude of the last $m-k$ entries of the wanted Ritz vectors W_m^1 (see (4.2)) is small enough. The next theorem, which extends Theorem 3.2 of [17], shows that $\|W_m^{21}\|$ is proportional to the estimated residual at step k.

THEOREM 4.1. *Let $\mathcal{A}U_k + F_k = U_k H_k + h_{k+1,k} u_{k+1} e_k^T$ be a k-step inexact Arnoldi decomposition, where the Schur decomposition of H_k is $H_k = W_k T_k W_k^*$. Let $m-k$ additional inexact Arnoldi steps be performed, giving $\mathcal{A}U_m + F_m = U_m H_m + h_{m+1,m} u_{m+1} e_m^T$. Let $R_k = (\mathcal{A}U_k + F_k - U_k H_k) W_k = h_{k+1,k} u_{k+1} e_k^T W_k$ be the estimated residual at Arnoldi step k. Given the Schur decomposition of H_m in* (4.2)*, then*

$$\frac{\|R_k\|}{\|R_k\| + \|\mathcal{S}_m\|} \le \|W_m^{21}\| \le \frac{\|R_k\|}{\operatorname{sep}(T_k, T_m^{22})}, \tag{4.6}$$

where $\mathcal{S}_m$ is the Sylvester operator $G \to \mathcal{S}_m(G) : T_m^{22} G - G T_k$, $\operatorname{sep}(T_k, T_m^{22}) = \min_{\|G\|=1} \|\mathcal{S}_m(G)\|$, and $\|\mathcal{S}_m\| = \max_{\|G\|=1} \|\mathcal{S}_m(G)\|$.

Proof. We need only to prove the lower bound, as the upper bound is established in Theorem 3.2 of [17]. The estimated residual norm at step k is

$$\|R_k\| = \|h_{k+1,k} u_{k+1} e_k^T W_k\| = h_{k+1,k} \|e_k^T W_k\| = h_{k+1,k}. \tag{4.7}$$

Consider the first block of the Schur decomposition of H_m (4.2):

$$
\begin{aligned}
(4.8)\quad & \left\| H_m \begin{pmatrix} W_k \\ 0 \end{pmatrix} - \begin{pmatrix} W_k \\ 0 \end{pmatrix} T_k \right\| \\
&= \left\| \begin{pmatrix} H_k & H_m^{12} \\ h_{k+1,k} e_1 e_k^T & H_m^{22} \end{pmatrix} \begin{pmatrix} W_k \\ 0 \end{pmatrix} - \begin{pmatrix} W_k \\ 0 \end{pmatrix} T_k \right\| \\
&= \left\| \begin{pmatrix} H_k \\ h_{k+1,k} e_1 e_k^T \end{pmatrix} W_k - \begin{pmatrix} W_k \\ 0 \end{pmatrix} T_k \right\| = \left\| \begin{pmatrix} H_k W_k - W_k T_k \\ h_{k+1,k} e_1 e_k^T W_k \end{pmatrix} \right\| \\
&= \left\| \begin{pmatrix} 0 \\ h_{k+1,k} e_1 e_k^T W_k \end{pmatrix} \right\| = h_{k+1,k} \| e_k^T W_k \| = h_{k+1,k},
\end{aligned}
$$

so that the first line of (4.8) also equals the estimated residual norm. Since W_m is unitary, it follows that with the first expression in the last line of (4.8), we have

$$
\begin{aligned}
(4.9)\quad \|R_k\| &= \left\| W_m^* \left(H_m \begin{pmatrix} W_k \\ 0 \end{pmatrix} - \begin{pmatrix} W_k \\ 0 \end{pmatrix} T_k \right) \right\| \\
&= \left\| \begin{pmatrix} (W_m^{11})^* & (W_m^{21})^* \\ (W_m^{12})^* & (W_m^{22})^* \end{pmatrix} \begin{pmatrix} 0 \\ h_{k+1,k} e_1 e_k^T W_k \end{pmatrix} \right\| \\
&= \left\| \begin{pmatrix} h_{k+1,k} (W_m^{21})^* e_1 e_k^T W_k \\ h_{k+1,k} (W_m^{22})^* e_1 e_k^T W_k \end{pmatrix} \right\|.
\end{aligned}
$$

On the other hand, using the Schur decomposition of H_m, we also have

$$
\begin{aligned}
(4.10)\quad \|R_k\| &= \left\| W_m^* \left(H_m \begin{pmatrix} W_k \\ 0 \end{pmatrix} - \begin{pmatrix} W_k \\ 0 \end{pmatrix} T_k \right) \right\| \\
&= \left\| T_m W_m^* \begin{pmatrix} W_k \\ 0 \end{pmatrix} - W_m^* \begin{pmatrix} W_k \\ 0 \end{pmatrix} T_k \right\| \\
&= \left\| \begin{pmatrix} T_m^{11} & T_m^{12} \\ 0 & T_m^{22} \end{pmatrix} \begin{pmatrix} (W_m^{11})^* W_k \\ (W_m^{12})^* W_k \end{pmatrix} - \begin{pmatrix} (W_m^{11})^* W_k \\ (W_m^{12})^* W_k \end{pmatrix} T_k \right\| \\
&= \left\| \begin{pmatrix} T_m^{11} (W_m^{11})^* W_k - (W_m^{11})^* W_k T_k + T_m^{12} (W_m^{12})^* W_k \\ T_m^{22} (W_m^{12})^* W_k - (W_m^{12})^* W_k T_k \end{pmatrix} \right\|.
\end{aligned}
$$

Using the upper block from (4.9) and lower block from (4.10), we have

$$
\begin{aligned}
(4.11)\quad \|R_k\| &= \left\| \begin{pmatrix} h_{k+1,k} (W_m^{21})^* e_1 e_k^T W_k \\ T_m^{22} (W_m^{12})^* W_k - (W_m^{12})^* W_k T_k \end{pmatrix} \right\| \\
&\le \| h_{k+1,k} (W_m^{21})^* e_1 e_k^T W_k \| + \| T_m^{22} (W_m^{12})^* W_k - (W_m^{12})^* W_k T_k \| \\
&\le h_{k+1,k} \| e_k^T W_k \| \| e_1^T W_m^{21} \| + \| \mathcal{S}_m \| \| (W_m^{12})^* W_k \| \\
&= \|R_k\| \| e_1^T W_m^{21} \| + \| \mathcal{S}_m \| \| W_m^{12} \| \le \|R_k\| \| W_m^{21} \| + \| \mathcal{S}_m \| \| W_m^{21} \|.
\end{aligned}
$$

Note that in the last line of (4.11), $\|(W_m^{12})^* W_k\| = \|W_m^{12}\| = \|W_m^{21}\|$ (see Theorem 2.6.1 in Golub and van Loan [19]). The lower bound in (4.6) is thus established. ∎

As observed above, for a given k-step inexact Arnoldi decomposition with small $\|F_k\|$, the error matrix $\|F_{k+1:m}\|$ associated with the upcoming $m-k$ inexact Arnoldi steps must be controlled appropriately to make sure that $U_m W_m^1$ is not obviously contaminated after these steps. In particular, $\|f_{k+1}\|$ cannot be too big. The following theorem gives an upper bound of $\|f_{k+1}\|$.

THEOREM 4.2. *Given $\epsilon_1 > 0$, suppose we have a k-step inexact Arnoldi decomposition $\mathcal{A}U_k + F_k = U_k H_k + h_{k+1,k} u_{k+1} e_k^T$, where $\|F_k\| \le \epsilon_1$. Let $\mathcal{S}_{k+1}$ be defined as in Theorem* 4.1. *Then for the next Arnoldi step,*

$$\|f_{k+1}\| \le \left(1 + \frac{\|\mathcal{S}_{k+1}\|}{\|R_k\|}\right)(\epsilon_1 + \epsilon_2) \tag{4.12}$$

is a necessary condition to make $\|(\mathcal{A}U_{k+1}W_{k+1}^1 - U_{k+1}W_{k+1}^1 T_{k+1}^{11}) - R_{k+1}\| \le \epsilon_2$.

Proof. Let $m = k+1$. We have the following estimate of the difference between the computed and true residual:

$$\begin{aligned} \|(\mathcal{A}U_{k+1}W_{k+1}^1 - U_{k+1}W_{k+1}^1 T_{k+1}^{11}) - R_{k+1}\| &= \|F_{k+1}W_{k+1}^1\| \\ &= \|F_k W_{k+1}^{11} + f_{k+1}W_{k+1}^{21}\| \ge \|f_{k+1}W_{k+1}^{21}\| - \|F_k W_{k+1}^{11}\| \\ &\ge \|f_{k+1}\|\|W_{k+1}^{21}\| - \|F_k\|\|W_{k+1}^{11}\| \ge \|f_{k+1}\|\frac{\|R_k\|}{\|R_k\| + \|\mathcal{S}_{k+1}\|} - \epsilon_1. \end{aligned} \tag{4.13}$$

Note that $\|f_{k+1}W_{k+1}^{21}\| = \|f_{k+1}\|\|W_{k+1}^{21}\|$ because f_{k+1} and W_{k+1}^{21} are, respectively, a column vector and row vector, and $\|W_{k+1}^{11}\| \le \|W_{k+1}^1\| = 1$. It follows immediately that (4.13) is bigger than ϵ_2 if $\|f_{k+1}\| > (1 + \frac{\|\mathcal{S}_{k+1}\|}{\|R_k\|})(\epsilon_1 + \epsilon_2)$. ∎

A practical choice would be $\epsilon_1 = \epsilon_2 \equiv \epsilon$. Using the upper bound of $\|W_m^{21}\|$ in Theorem 4.1, we can also show that $\|f_{k+1}\| \le \frac{\mathrm{sep}(T_m^{22}, T_k)}{\|R_k\|}\epsilon$ is *sufficient* to make $\|(\mathcal{A}U_{k+1}W_{k+1}^1 - U_{k+1}W_{k+1}^1 T_{k+1}^{11}) - R_{k+1}\| \le 2\epsilon$.

However, the bounds of $\|f_{k+1}\|$ in the necessary and sufficient conditions might severely overestimate and underestimate, respectively, the actual allowable error in the $(k+1)$th Arnoldi step. In fact, $\mathrm{sep}(T_{k+1}^{22}, T_k)$ and $\|\mathcal{S}_{k+1}\|$ are analogous to the smallest and the largest singular values of the Sylvester operator $\mathcal{S}_{k+1}$. The necessary condition is generally too weak, as an obviously smaller $\|f_{k+1}\|$ may still not suffice to keep the approximate invariant subspace from being contaminated. On the other hand, the sufficient condition might be overly conservative, giving excessively small tolerance for the linear system $Ay = Bu_{k+j+1}$ $(0 \le j \le m-k-1)$ and leading to unnecessary extra inner iterations. To give a practical estimate of the allowable $\|F_{k+1:m}\|$, [17] substitutes $\min|\lambda(T_k) - \lambda(T_m^{22})|$ for $\mathrm{sep}(T_m^{22}, T_k)$, which is difficult to estimate. Since $\min|\lambda(T_k) - \lambda(T_m^{22})| > \mathrm{sep}(T_m^{22}, T_k)$ for nonnormal $\mathcal{A}$, this substitution essentially gives a less conservative estimate.

A better estimate should be a trade-off between these two conditions. Theorem 3.2 of [17] uses $\|T_m^{22}(W_m^{12})^* W_k - (W_m^{12})^* W_k T_k\| \ge \mathrm{sep}(T_m^{22}, T_k)\|(W_m^{12})^* W_k\|$, whereas Theorem 4.1 above applies $\|T_m^{22}(W_m^{12})^* W_k - (W_m^{12})^* W_k T_k\| \le \|\mathcal{S}_m\|\|(W_m^{12})^* W_k\|$. Therefore, a more accurate estimate can be obtained by replacing the lower bound $\mathrm{sep}(T_m^{22}, T_k)$ and upper bound $\|\mathcal{S}_m\|$ by

$$\frac{\|T_m^{22}(W_m^{12})^* W_k - (W_m^{12})^* W_k T_k\|}{\|(W_m^{12})^* W_k\|} = \frac{\|T_m^{22}(W_m^{12})^* - (W_m^{12})^* H_k\|}{\|W_m^{12}\|}, \tag{4.14}$$

which takes into account the actual effect of $\mathcal{S}_m$ on $(W_m^{12})^* W_k$. Here we use the fact that $H_k = W_k T_k W_k^*$ is a Schur decomposition.

The above strategy gives a theoretically more accurate estimate of $\|F_{k+1:m}\|$. However, like the estimate $\min|\lambda(T_k) - \lambda(T_m^{22})|$ in [17], it depends on the Schur decomposition of H_m, which is not available at step k. A practical (heuristic) solution is to use the decomposition of H_m from the previous IRA cycle and H_k of the current cycle. Specifically, suppose at the beginning of the ith IRA cycle, we have $\mathcal{A}U_k^{(i)} +$

$F_k^{(i)} = U_k^{(i)} H_k^{(i)} + h_{k+1,k}^{(i)} u_{k+1}^{(i)} e_k^T$. Then we define

$$\sigma_{est}^{(i)} \equiv \frac{\|T_m^{22(i-1)}(W_m^{12(i-1)})^* - (W_m^{12(i-1)})^* H_k^{(i)}\|}{\|W_m^{12(i-1)}\|}, \tag{4.15}$$

which is very easy to compute. Note that $H_k^{(i)} = \tilde{H}_k^{(i-1)}$ if the exact shift strategy is used; see (2.4). Substituting $\sigma_{est}^{(i)}$ for $\mathrm{sep}(T_m^{22(i)}, T_k^{(i)})$ in the relaxation strategy (3.13) in [17], we have the following *heuristic* estimate of the allowable errors:

$$\|f_j^{(i)}\| \le \frac{\epsilon}{2k} \qquad (i = 1, 1 \le j \le m) \tag{4.16}$$

and

$$\|f_{k+j+1}^{(i)}\| \le \frac{\epsilon}{2(m-k)} \frac{\sigma_{est}^{(i)}}{\|R_k^{(i)}\|} \qquad (i > 1, 0 \le j \le m-k-1),$$

which hopefully can also lead to the generation of an approximate invariant subspace $U_m^{(i)}$ satisfying $\|\mathcal{A}U_m^{(i)} - U_m^{(i)} H_m^{(i)} - h_{m+1,m} u_{m+1} e_m^T\| \le \epsilon$.

Remark. To the best of our knowledge, given a k-step inexact Arnoldi decomposition with small $\|F_k\|$, none of the existing *practical* (computable) estimates of allowable $\|F_{k+1:m}\|$ can theoretically guarantee that the desired approximate invariant subspace $U_m W_m^1$ will not be contaminated after (m–k) inexact Arnoldi steps. The estimate in [28] for the unrestarted Arnoldi method involves the distance between the desired Ritz value extracted from H_k and the rest of the spectrum of H_k, assuming that the computed eigenvalue residual at step k is already small enough, which might not be the case (see section 3.1 of [28] for the explicit formula); reference [17] uses $\min|\lambda(T_k^{(i)}) - \lambda(T_m^{22(i-1)})|$ in place of $\mathrm{sep}(T_k^{(i)}, T_m^{22(i)})$, which is replaced by $\sigma_{est}^{(i)}$ in our new estimate. We will compare the new estimate with that in [17] in section 5.

We also point out that $\|F_m\|$ should be properly scaled. In fact, as $\mathcal{A}U_m + F_m = U_m H_m + h_{m+1,m} u_{m+1} e_m^T$, the *relative* quantity $\frac{\|F_m\|}{\|\mathcal{A}U_m\|}$ should be used to measure the magnitude of errors, especially if $\|\mathcal{A}U_m\|$ is *not* moderate. Specifically, at the $(k+j+1)$th Arnoldi step, the linear system $Ay = Bu_{k+j+1}$ needs to be solved inexactly. The relative error $\frac{\|f_{k+j+1}\|}{\|\mathcal{A}u_{k+j+1}\|} = \frac{\|y - A^{-1}Bu_{k+j+1}\|}{\|A^{-1}Bu_{k+j+1}\|}$ is not available as we do not have $A^{-1}Bu_{k+j+1}$. A reasonable and convenient substitute is the relative residual norm of this linear system $\frac{\|Ay - Bu_{k+j+1}\|}{\|Bu_{k+j+1}\|}$. For our inexact IRA method, we require this quantity to be bounded above by the new estimate in (4.16).

Finally, we note that inexact IRA is most suitable when a small number of eigenpairs are wanted. This algorithm computes all desired eigenpairs simultaneously, and the maximum allowable inner solves errors are inversely proportional to the eigenresidual norm of the whole desired approximate invariant subspace. Therefore, the inner solve errors need to be kept small until the wanted eigenpairs that converge most slowly are resolved to moderate accuracy. In scenarios where many (say, hundreds of) eigenpairs need to be computed, it may be more practical to use eigensolvers such as the Jacobi–Davidson method that compute one eigenpair at a time.

5. Numerical experiments. We present and discuss the results of numerical experiments in this section, showing the effectiveness of the new tuning strategy, subspace recycling, and the new relaxation strategy. The following issues are addressed:

1. We show that the tuning strategy constructed using solution vectors obtained from previous Arnoldi steps works as Theorem 3.4 describes: a proper use of tuning in Phase I of Algorithm 2 gives a minimum residual solution y_1 for

TABLE 5.1
Parameters used to solve the test problems.

	k_w	k	m	σ (σ_1)	σ_2	τ	ϵ	p_1	p_2	l
Prob 1	8	8	12	0	–	1×10^{-12}	2×10^{-11}	10	10	5
Prob 2	3	4	9	−0.0325	0.125	1×10^{-9}	2×10^{-10}	10	10	4
Prob 3(a)	5	7	13	0	–	5×10^{-11}	5×10^{-9}	0	25	6
Prob 3(b)	5	9	15	0	−0.46	5×10^{-11}	5×10^{-9}	0	25	6
Prob 4(a)	5	7	13	0	–	5×10^{-11}	5×10^{-9}	0	40	6
Prob 4(b)	5	9	15	0	−0.24	5×10^{-11}	5×10^{-9}	0	40	6

which $\frac{\|Bu_{k+j+1}^{(i)} - Ay_1\|}{\|Bu_{k+j+1}^{(i)}\|} = O(s_p^{(i,l)}) \ll 1$, and therefore the correction equation can be solved with a less stringent relative tolerance $\frac{\delta\|Bu_{k+j+1}^{(i)}\|}{\|Bu_{k+j+1}^{(i)} - Ay_1\|} \gg \delta$, no matter whether δ is a fixed or relaxed relative tolerance for $Ay = Bu_{k+j+1}^{(i)}$. The new tuning strategy is compared with the original tuning strategy in [17].

2. We compare inexact IRA methods with nonrelaxed (fixed) tolerances $\delta_f = \frac{\epsilon}{2k}$, where ϵ and k are given in Table 5.1 (this δ_f is used in [17] for inexact IRA with a fixed tolerance for the inner solve) and relaxed tolerances δ_r given by either the original estimate in [17] or the new estimate in (4.16). The accuracy of the two estimates is discussed based on the numerical results.
3. We show that further reduction of inner iteration counts can be achieved at little cost by proper subspace recycling.

We first explain the stopping criterion for the inexact IRA method. Suppose at the beginning of the ith IRA cycle we have $\mathcal{A}U_k^{(i)} + F_k^{(i)} = U_k^{(i)}H_k^{(i)} + h_{k+1,k}^{(i)}u_{k+1}^{(i)}e_k^T$. Let $(\theta_j^{(i)}, v_j^{(i)})$ $(1 \le j \le k)$ be a Ritz pair, i.e., an eigenpair of $H_k^{(i)}$. Postmultiplying the above equation by $v_j^{(i)}$, we have

$$\mathcal{A}(U_k^{(i)}v_j^{(i)}) - \theta_j^{(i)}(U_k^{(i)}v_j^{(i)}) - (h_{k+1,k}^{(i)}v_{kj}^{(i)})u_{k+1}^{(i)} = -F_k^{(i)}v_j^{(i)}, \tag{5.1}$$

where $v_{kj}^{(i)}$ is the kth (last) entry of $v_j^{(i)}$. Here $U_k^{(i)}v_j^{(i)}$ is an approximate eigenvector of $\mathcal{A}$, $\mathcal{A}(U_k^{(i)}v_j^{(i)}) - \theta_j^{(i)}(U_k^{(i)}v_j^{(i)})$ is the true eigenvalue residual, and $(h_{k+1,k}^{(i)}v_{kj}^{(i)})u_{k+1}^{(i)}$ is the estimated residual. As the magnitude of errors has been kept under control to guarantee that the true residual is close enough to the estimated one, we know that

$$\left|\frac{h_{k+1,k}^{(i)}v_{kj}^{(i)}}{\theta_j^{(i)}}\right| \approx \frac{\|\mathcal{A}(U_k^{(i)}v_j^{(i)}) - \theta_j^{(i)}(U_k^{(i)}v_j^{(i)})\|}{|\theta_j^{(i)}|}. \tag{5.2}$$

By checking that the computed residual (the left-hand side of (5.2)) is smaller than a prescribed tolerance τ, we have confidence that the true residual (right-hand side of (5.2)) is also approximately equal to or smaller than τ. Using estimated residuals avoids the overhead of computing $\frac{\|A(U_k^{(i)}v_j^{(i)}) - \theta_j^{(i)}B(U_k^{(i)}v_j^{(i)})\|}{|\theta_j^{(i)}|}$ to evaluate the quality of approximate eigenvectors during the IRA process. We use the relative residual norm here because the dominant eigenpairs of $\mathcal{A} = A^{-1}B$ are computed, and therefore θ is generally not small.

Another issue is that k does not have to be equal to the number of desired eigenpairs k_w. One can choose a slightly bigger k for the IRA method, and only test $|(\theta_j^{(i)})^{-1}h_{k+1,k}^{(i)}v_{kj}^{(i)}|$ in (5.2) for $1 \le j \le k_w$. Our experience is that for fixed $m-k$ (the number of Arnoldi steps in each *restarted* IRA cycle), more often than not, this

choice of k reduces the number of IRA cycles. This idea, called "thick restarting," has been studied for the generalized Davidson method [35]. We speculate that thick restarting for IRA makes the unwanted Ritz values more separated from the desired eigenvalues; therefore it is less likely for the filter polynomial to damp the desired eigenvector components during the restart.

Four test problems are used in our numerical experiments. The first is SHERMAN5 from MatrixMarket [23], a real matrix of order 3312 arising from oil reservoir modeling. We use the shift-invert operator $\mathcal{A} = A^{-1}$ (with $B = I$) to detect some eigenvalues closest to zero. The inner solve is done with ILU preconditioning with drop tolerance 0.008 given by MATLAB's `ilu` function. This example is used in [17] to show the effectiveness of the tuning and the relaxation strategy therein.

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

Problems 3 and 4 arise 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 [9, 10]. The domain is $[-1, L]\times[-1, 1]$ with $[-1, 0]\times[-1, 0]$ cut out, where $L = 15$ in problem 3 and $L = 23$ in problem 4; the Reynolds numbers are 600 and 1200, respectively. Let u and v be the horizontal and vertical component of the velocity, let p be the pressure, and let ν be the viscosity. The boundary conditions are as follows:

$$
\begin{aligned}
& u = 4y(1-y),\ v = 0 \text{ (parabolic inflow)} && \text{on } x = -1,\ y \in [0,1]; \\
& \nu\frac{\partial u}{\partial x} - p = 0,\ \frac{\partial v}{\partial y} = 0 \text{ (natural outflow)} && \text{on } x = L,\ y \in [-1,1]; \\
& u = v = 0 \text{ (no-slip)} && \text{on all other boundaries.}
\end{aligned} \tag{5.3}
$$

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 two problems are of order 72867 and 110371, respectively. We use the least squares commutator preconditioner [11] for the inner solves. For both problems, we try shift-invert (subproblem (a)) and Cayley transformation (subproblem (b)) to detect a small number of critical eigenvalues.

For completeness, the parameters used in the solution of each test problem are given in Table 5.1. These parameters are chosen to deliver approximate eigenpairs of adequate accuracies and show representative behavior of each solution strategy.

1. k_w, k, m: We use the IRA method to compute k_w eigenpairs; m and k are the order of the Arnoldi decomposition before and after the implicit restart.
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. τ: We stop the IRA method if the estimated residual in (5.2) is smaller than τ for all k_w desired approximate eigenpairs.
4. ϵ: The small quantity used in (4.16) to estimate the allowable tolerances for the linear systems.
5. p_1, p_2: p_1 harmonic Ritz vectors corresponding to harmonic Ritz values of smallest magnitude and p_2 dominant Ritz vectors are used for subspace recycling.
6. l: Solutions vectors from l preceding IRA cycles are used for the tuning in Phase I of Algorithm 2.

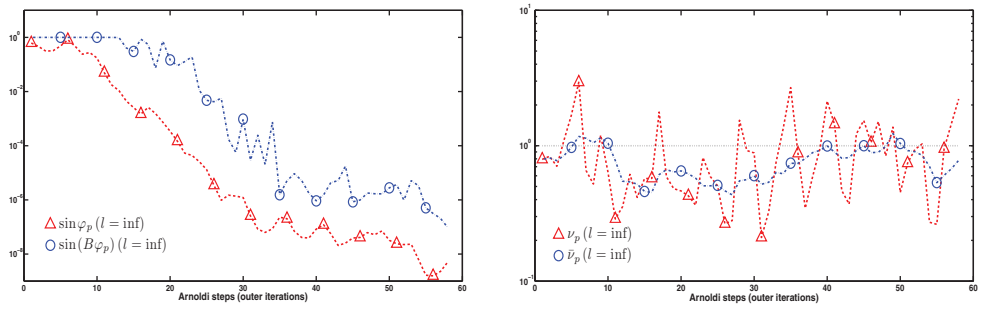

FIG. 5.1. *Linear decrease of* $\sin\varphi_p$ *for problem* 3(a). *Left:* $\sin\varphi_p$ *and* $\sin(B\varphi_p)$. *Right:* ν_p *and its geometric average over six consecutive steps.*

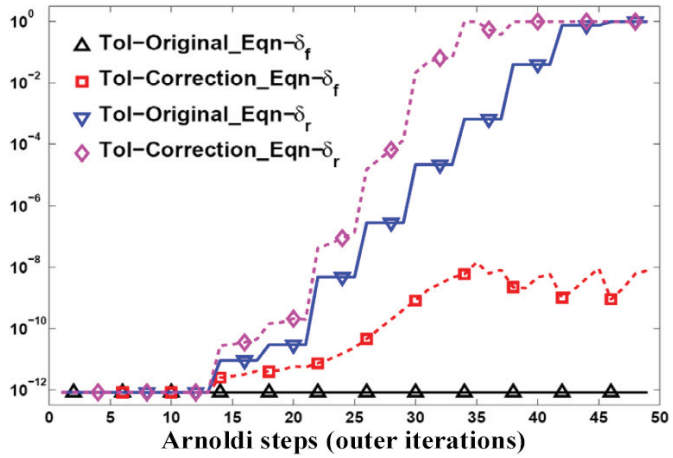

FIG. 5.2. *Problem* 1: *Relative tolerances for the original systems and correction equations.*

We begin by exploring the major quantities discussed in Theorems 3.2 and 3.3, using problem 3(a) as a benchmark problem; results for the other test problems discussed in this section were similar. The inner solve is done with preconditioned GMRES to the fixed relative tolerance 3.5×10^{-10}. From the first restarted IRA cycle, at every Arnoldi step, we choose the solution vectors obtained in *all* previous steps for tuning ($l = \text{inf}$; see (3.2)). The left part of Figure 5.1 plots $\sin\varphi_p \equiv \sin\angle(u_{k+j+1}^{(i)}, \mathcal{U}_p^{(i,l)})$ ($\triangle$) and $\sin(B\varphi_p) \equiv \sin\angle(Bu_{k+j+1}^{(i)}, B\mathcal{U}_p^{(i,l)})$ ($\bigcirc$) against the Arnoldi steps, and the right part of the figure shows ν_p and $\bar{\nu}_p = (\prod_{s=0}^{5}\nu_{p-s})^{1/6}$ (the geometric average of ν_p over six steps). Here, note that $m - k = 6$ Arnoldi steps are performed in each restarted IRA cycle, so that this geometric average is an estimate of the variation of $\sin\angle(u_{k+j+1}^{(i)}, \mathcal{U}_p^{(i,l)})$ in one IRA cycle. It can be seen from the figure that, generally speaking, both $\sin\varphi_p$ and $\sin(B\varphi_p)$ decrease linearly with p, although $\sin(B\varphi_p)$ is roughly one to two orders of magnitude larger. The scalar ν_p fluctuates from step to step, and it can also be seen that cases where it is greater than one (for example, midway between Arnoldi steps 30 and 40) coincide with increases in $\sin(B\varphi_p)$. But ν_p is smaller than 1 at a majority of steps, as is its geometric average.

Figure 5.2 plots the relative tolerances δ for the original systems $Ay = Bu_{k+j+1}^{(i)}$ (solid lines) and the derived relative tolerances $\frac{\delta\|Bu_{k+j+1}^{(i)}\|}{\|Bu_{k+j+1}^{(i)} - Ay_1\|}$ for the correction

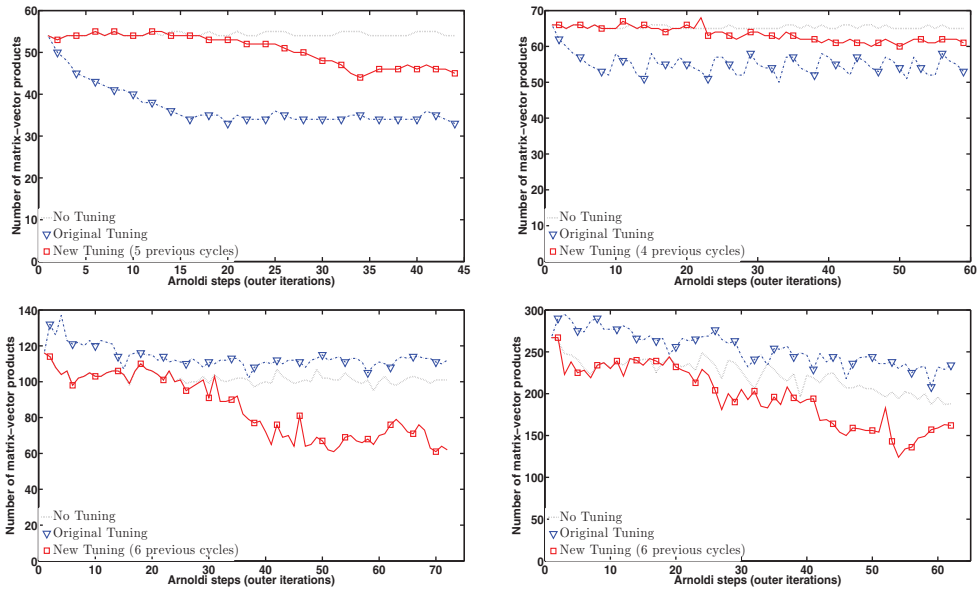

FIG. 5.3. *Performance of different strategies with fixed tolerances of inner solves for problems* 1, 2, 3(a), *and* 4(a).

equations $Az = Bu_{k+j+1}^{(i)} - Ay_1$ (dashed lines) for problem 1. The curves are as follows:

- $\triangle$ Tol-Original_Eqn-δ_f and $\square$ Tol-Correction_Eqn-δ_f: The fixed relative tolerance $\delta_f = \frac{\epsilon}{2k} \approx 10^{-12}$ for the original system $Ay = Bu_{k+j+1}^{(i)}$ and the derived relative tolerance $\frac{\delta_f \|Bu_{k+j+1}^{(i)}\|}{\|Bu_{k+j+1}^{(i)} - Ay_1\|}$ for the correction equation $Az = Bu_{k+j+1}^{(i)} - Ay_1$.
- $\triangledown$ Tol-Original_Eqn-δ_r and $\diamondsuit$ Tol-Correction_Eqn-δ_r: The relaxed relative tolerances δ_r estimated by (4.16) for $Ay = Bu_{k+j+1}^{(i)}$ and the derived relative tolerance $\frac{\delta_r \|Bu_{k+j+1}^{(i)}\|}{\|Bu_{k+j+1}^{(i)} - Ay_1\|}$ for $Az = Bu_{k+j+1}^{(i)} - Ay_1$.

Figure 5.2 corroborates the property of the two-phase algorithm described in Theorem 3.4. Specifically, after Phase I of Algorithm 2, we get a good approximate solution y_1 for which the relative residual norm $\frac{\|Bu_{k+j+1}^{(i)} - Ay_1\|}{\|Bu_{k+j+1}^{(i)}\|} = O(s_p^{(i,l)}) \ll 1$, and therefore the derived relative tolerance of the correction equation $\frac{\delta \|Bu_{k+j+1}^{(i)}\|}{\|Bu_{k+j+1}^{(i)} - Ay_1\|} \gg \delta$.

The reduction of inner iterations by the two-phase algorithm (with the new tuning used in Phase I of Algorithm 2) can be seen from Figure 5.3, where the inner iteration counts required by three different strategies for solving $Ay = Bu_{k+j+1}^{(i)}$ are plotted against the Arnoldi steps:

- "No Tuning" (dotted line): Solve the original systems $Ay = Bu_{k+j+1}^{(i)}$ by preconditioned GMRES to the fixed tolerance $\delta_f = \frac{\epsilon}{2k}$ without any enhancements.
- "Original Tuning" ($\triangle$, solid line): Solve $Ay = Bu_{k+j+1}^{(i)}$ by GMRES with the original version of tuning in [17] to the fixed tolerance δ_f (note that the original tuning needs to be applied at each GMRES step so that the eigenvalues of the preconditioned system matrix can be clustered and hence

inner iteration counts can be reduced; this effect cannot be realized by trying the two-phase approach with this tuning strategy).

- "New Tuning (five previous cycles)" (□, dashed line): Solve $Ay = Bu_{k+j+1}^{(i)}$ by the two-phase algorithm to the fixed tolerance δ_f; in the first phase, the new tuning is constructed using solution vectors from the current and five previous IRA cycles.

For each test problem, we choose l (the number of previous IRA cycles) such that the use of our tuning with a larger l does not obviously further reduce the total inner iteration counts. Clearly, compared to the "No Tuning" strategy, the two-phase algorithm (□, with the new tuning used in Phase I only) requires fewer inner iterations due to the larger relative tolerances for the correction equations.

It can be seen that there are some cases where the original tuning strategy is superior to the new one (see the plots at the top of Figure 5.3) and others where the original tuning is ineffective (bottom plots of Figure 5.3). The distinct merits of the two approaches can be explained as follows. Given some untuned preconditioner P, let $\mathbb{P}$ be the original tuned version defined in [17]. It is shown in [17] that the preconditioned operator $A\mathbb{P}^{-1}$ tends to have more favorable eigenvalue clustering than AP^{-1}, especially if P is not a strong preconditioner. For problems 1 and 2, we have seen that the use of tuning (of the ILU preconditioner) produces preconditioned coefficient matrices with considerably tighter clustering of eigenvalues than for the untuned preconditioner. This improves the performance of GMRES. Moreover, it appears that tuning forces some small eigenvalues to move away from the origin, and as a result, the initial latency of preconditioned GMRES is significantly shortened. On the other hand, with the new tuning strategy, used with the two-phase method of Algorithm 2, there is still a long initial latency for solving the correction equation by GMRES with the *untuned* perconditioner P, despite the larger tolerance allowed. Moreover, as observed above, the new tuning does not improve performance for solving the correction equation.

However, the original tuning is not always effective for this purpose. As Figure 5.3 shows, for problems 3 and 4, the use of the original tuning (△) requires even more inner iterations than the untuned preconditioner (dotted line) does, whereas the new tuning (□) reduces the inner iteration counts considerably. In these cases, the linear solvers make use of the least squares commutator preconditioner [11, 12]; this is a strong preconditioner for which the preconditioned system matrix AP^{-1} has most eigenvalues clustered around 1 and only a small number of outliers [12]. In this case, we find that the clustering of eigenvalues is not significantly improved by the original tuning (indeed, it may be compromised), and the performance of preconditioned GMRES is not improved by tuning. On the other hand, as shown in section 3.2, the improved performance of the new tuning method comes from the fact that the right-hand side of the system being solved is nearly an eigenvector of the preconditioned operator together with the relaxed tolerance for solving the correction equation in Algorithm 2.

In addition to the use of the new tuning strategy, the overall performance of the two-phase algorithm can be improved by efficient solution of the correction equation $Az = Bu_{k+j+1}^{(i)} - Ay_1$. This can be performed using the GCRO-DR algorithm discussed in section 3.3. Compared to the regular GMRES solve, the use of subspace recycling achieves improved performance because it deflates some smallest and largest (in magnitude) eigenvalues of the coefficient matrix and thus helps reduce the initial latencies of inner iterations. Also, as observed in section 3.3, the recycled subspaces obtained from one correction equation can be applied directly to the solution of the

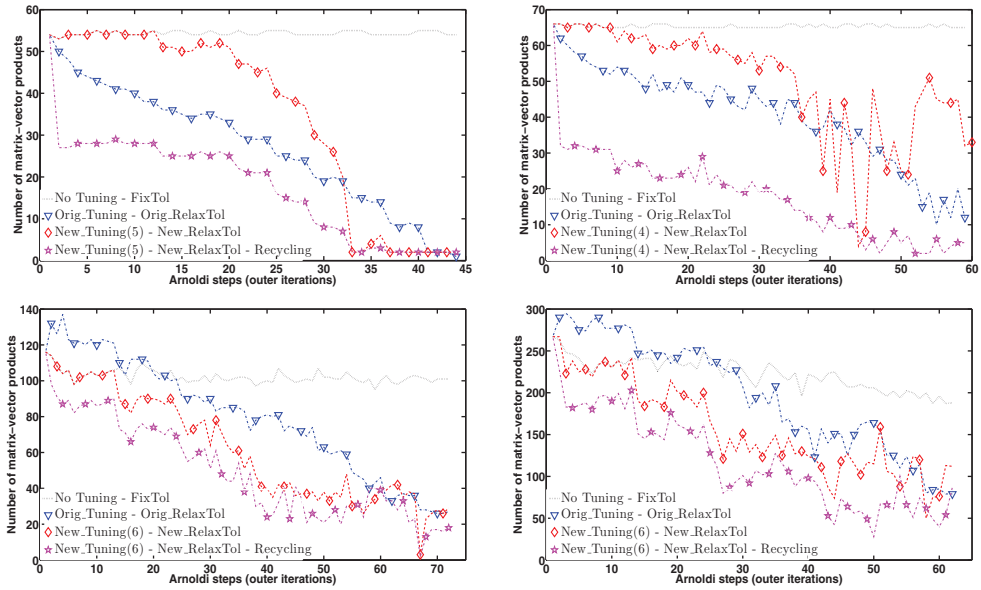

FIG. 5.4. *Performance of different strategies with relaxed tolerances of inner solves for problems* 1, 2, 3(a), *and* 4(a).

next equation, because the preconditioned system matrix is identical for the correction equations in all Arnoldi steps. This makes subspace recycling very cheap to use.

Moreover, further performance improvement can be achieved by the use of the relaxation strategy: As section 4 shows, the allowable tolerances for $Ay = Bu_{k+j+1}^{(i)}$ are inversely proportional to the current eigenvalue residual norm. Therefore as the IRA method proceeds and converges to the desired invariant subspace, the relaxed tolerances keep increasing. Figure 5.4 shows the inner iteration counts required by four strategies for solving $Ay = Bu_{k+j+1}^{(i)}$:

- "No Tuning-FixTol" (dotted line): Solve the original systems $Ay = Bu_{k+j+1}^{(i)}$ with preconditioned GMRES to the *fixed* tolerance $\delta_f = \frac{\epsilon}{2k}$. This performance of this strategy is already given in Figure 5.3; it is shown again to illustrate the performance improvement obtained by the following advanced strategies.
- "Orig_Tuning-Orig_RelaxTol" ($\triangledown$, solid line): Solve $Ay = Bu_{k+j+1}^{(i)}$ by GMRES with the original tuning to the *relaxed* tolerances δ_r given by the original estimate.
- "New_Tuning(5)-New_RelaxTol" ($\diamondsuit$, dashed line): Solve $Ay = Bu_{k+j+1}^{(i)}$ by the two-phase strategy to the new estimated tolerances δ_r in (4.16); tuning is constructed using solution vectors from the current and five previous IRA cycles.
- "New_Tuning(5)-New_RelaxTol-Recycling" ($\bigstar$, dashed line): Solve $Ay = Bu_{k+j+1}^{(i)}$ by by the two-phase strategy to the new estimated tolerances δ_r; in addition, subspace recycling is used to solve the correction equations.

One can see from Figure 5.4 that the relaxed tolerances help gradually reduce the inner iteration counts to small numbers (curves with $\triangledown$, $\diamondsuit$, and $\bigstar$). The effectiveness of subspace recycling is also clear: for problems 1 and 2, the use of this technique reduces

TABLE 5.2
Inner iteration counts for different solution strategy for each problem.

	No tuning	New tuning	New tuning new relaxation	New tuning new relaxation subspace recycling	New tuning original relaxation subspace recycling	Original tuning	Original tuning original relaxation
Prob 1	2391	2233	1531	765	795	1628	1194
Prob 2	3856	3771	3001	1052	1096	3250	2397
Prob 3(a)	7367	6153	4450	3584	3618	8189	5737
Prob 3(b)	10661	8079	7088	5295	5481	10576	7435
Prob 4(a)	13999	12203	9693	7294	7536	15738	12260
Prob 4(b)	19436	15954	12748	9745	10500	19552	14115

TABLE 5.3
Largest eigenvalue residual norms of eigenvector approximations computed by different solution strategies.

	No tuning	New tuning	New tuning new relaxation	New tuning new relaxation subspace recycling	New tuning original relaxation subspace recycling	Original tuning	Original tuning original relaxation
Prob 1	$1.398e{-12}$	$1.400e{-12}$	$1.754e{-12}$	$1.624e{-12}$	$1.468e{-12}$	$1.231e{-12}$	$1.407e{-12}$
Prob 2	$8.021e{-10}$	$8.021e{-10}$	$1.424e{-9}$	$8.956e{-10}$	$9.723e{-10}$	$2.169e{-9}$	$2.579e{-9}$
Prob 3(a)	$1.114e{-10}$	$1.116e{-10}$	$2.003e{-10}$	$1.770e{-10}$	$1.206e{-10}$	$1.310e{-10}$	$1.726e{-10}$
Prob 3(b)	$1.583e{-9}$	$1.578e{-9}$	$2.365e{-9}$	$5.619e{-9}$	$2.445e{-9}$	$1.692e{-9}$	$1.884e{-9}$
Prob 4(a)	$4.605e{-11}$	$1.152e{-10}$	$1.152e{-10}$	$8.660e{-11}$	$7.660e{-11}$	$1.192e{-10}$	$1.522e{-10}$
Prob 4(b)	$8.523e{-10}$	$8.517e{-10}$	$2.219e{-9}$	$1.939e{-9}$	$1.025e{-9}$	$8.701e{-10}$	$9.826e{-10}$

the inner iteration counts by 40%–50% in initial Arnoldi steps (compare curves with $\diamondsuit$ to those with $\bigstar$); for Problems 3 and 4, where the original tuning does not perform well (see Figure 5.3), subspace recycling still decreases the inner iteration counts of each linear solve by numbers commensurate to the dimensions of recycled subspaces.

Table 5.2 summarizes the total inner iteration counts needed for each strategy for solving $Ay = B^{(i)}_{k+j+1}$ arising in inexact IRA. Here, "New Tuning + New Relaxation + Subspace Recycling" and "Original Tuning + Original Relaxation" are the most efficient strategies in this paper and [17], respectively. Clearly, the most efficient approach is to combine the two-phase algorithm (with the new tuning), relaxation strategy, and subspace recycling. The largest eigenvalue residual norm, $\frac{\|Aw_j-\mu_j Bw_j\|}{\max\{1,|\mu_j|\}}$, of computed eigenpairs $(\mu_j, w_j)\,(1 \le j \le k_w)$ is given in Table 5.3. One can see that inexact IRA with any of these competing inner solve strategies delivers computed eigenpairs of approximately the same quality.

Finally, we discuss the two approaches assessing the allowable errors of inner solves in the Arnoldi steps. For all problems, we found that solution strategies with the new estimated allowable tolerances (4.16) require slightly smaller numbers of inner iterations than are needed for strategies with the original estimated tolerances. Table 5.2 shows that the new estimated tolerances help decrease the inner iteration counts by about 2%–5% (compare the "New Tuning + New Relaxation + Subspace Recycling" with "New Tuning + Original Relaxation + Subspace Recycling") when used with the two-phase strategy and subspace recycling. In fact, the new estimated

tolerances tend to be a small multiple (say, 2 to 10) of the original estimated ones in most IRA cycles for all test problems.

Some heuristic remarks can be made for the two estimations. First, the substitution of $\min|\lambda(T_k) - \lambda(T_m^{22})|$ for $\mathrm{sep}(T_m^{22}, T_k)$ in the original estimation seems reasonable, in the sense that the former is usually not obviously larger than the latter. In fact, in the setting of eigenvalue computation, we expect two basic properties to hold: (1) the desired Ritz vectors generated by the Rayleigh–Ritz procedure are not far from the "best approximation" in the subspace from which the Ritz vectors are extracted, and (2) a small eigenvalue residual of the desired approximate invariant subspace implies good eigenvector approximation. Here, the best approximation may refer to an eigenvector approximation in a given space obtained from the refined Rayleigh–Ritz method, an approximation in that space which minimizes the eigenvalue residual norm, or the orthogonal projection of the true eigenvector onto that space. These approximations are expected to be close to each other if the matrices involved are not highly nonnormal; see [36] for details. However, by analogy to the results in [37] and Chapter 2 of [21], both properties may not be true if $\mathrm{sep}(T_m^{22}, T_k)$ is considerably smaller than $\|T_m^{12}\|$ or $\min|\lambda(T_k) - \lambda(T_m^{22})|$ in our context. In the usual situations when the two properties hold, $\min|\lambda(T_k) - \lambda(T_m^{22})|$ is expected to be not much larger than $\mathrm{sep}(T_m^{22}, T_k)$. Second, numerical evidence help us understand why the new estimate tends to be slightly larger than the original estimate. In fact, note that $\min|\lambda(T_k) - \lambda(T_m^{22})|$ and $\max|\lambda(T_k) - \lambda(T_m^{22})|$ are the smallest and largest eigenvalue of the Sylvester operator $\mathcal{S}_m$ $(G \to \mathcal{S}_m(G) : T_m^{22}G - GT_k)$; see [36, page 17]. For the test problems with spectral transformation, it was consistently found that the largest eigenvalue of $\mathcal{S}_m$ is only about 10–100 times larger than the smallest eigenvalue of $\mathcal{S}_m$, as long as the shift is not too close to an eigenvalue of the matrix pair (A, B). As the quantity in (4.14) used in the new estimation is always between the two extreme eigenvalues of $\mathcal{S}_m$ in practice, it is not surprising that this quantity tends to be a small multiple of $\min|\lambda(T_k) - \lambda(T_m^{22})|$. The original estimated allowable tolerance seems reasonably accurate for the test problems.

6. Conclusions. We have studied an inexact implicitly restarted Arnoldi (IRA) method for solving generalized eigenvalue problems with shift-invert and Cayley transformations, with focus on a few strategies that help reduce the inner iteration counts. We present a new tuning strategy using the solution vectors from the current and previous IRA cycles, and discuss a two-phase algorithm involving a correction equation for which the tolerance can be considerably bigger than that for the original system. In addition, subspace recycling can be used easily for the correction equation to further reduce the inner iteration counts. We analyze the allowable errors of matrix-vector products performed in Arnoldi steps and propose an alternative estimate of relaxed tolerances for the original linear systems. Numerical experiments show that the combined use of these strategies lead to significant speedup of inner iterations.

Acknowledgment. We thank the referees for their insightful comments, which helped us improve the presentation and numerical experiments of this paper.

REFERENCES

[1] J. Berns-Müller and A. Spence, *Inexact Inverse Iteration and GMRES*, Technical report maths0507, University of Bath, Bath, UK, 2005

[2] J. Berns-Müller and A. Spence, *Inexact inverse iteration with variable shift for nonsymmetric generalized eigenvalue problems*, SIAM J. Matrix Anal. Appl., 28 (2006), pp. 1069–1082.

[3] J. Berns-Müller, I. G. Graham, and A. Spence, *Inexact inverse iteration for symmetric matrices*, Linear Algebra Appl., 416 (2006), pp. 389–413.
[4] A. Bouras and V. Frayssé, *A Relaxation Strategy for the Arnoldi Method in Eigenproblems*, Technical report TR/PA/00/16, CERFACS, Toulouse, France, 2000.
[5] A. Bouras and V. Frayssé, *A Relaxation Strategy for Inner-outer Linear Solvers in Domain Decomposition Methods*, Technical report TR/PA/00/16, CERFACS, Toulouse, France, 2000.
[6] A. Bouras and V. Frayssé, *Inexact matrix-vector products in Krylov methods for solving linear systems: A relaxation strategy*, SIAM J. Matrix Anal. Appl., 26 (2005), pp. 660–678.
[7] E. de Sturler, *Truncation strategies for optimal Krylov subspace methods*, SIAM J. Numer. Anal., 36 (1999), pp. 864–889.
[8] M. Eiermann, O. G. Ernst, and O. Schneider, *Analysis of acceleration strategies for restarted minimal residual methods*, J. Comput. Appl. Math. 123 (2000), pp. 261–292.
[9] H. C. Elman, A. R. Ramage, D. J. Silvester, and A. J. Wathen, *Incompressible Flow Iterative Solution Software Package, Version* 3.0, available online at http://www.cs.umd.edu/∼elman/ifiss.html.
[10] H. C. Elman, A. R. Ramage, and D. J. Silvester, *Algorithm IFISS* 886: *A Matlab toolbox for modelling incompressible flow*, ACM Trans. Math. Software, 33 (2007), Article 14.
[11] H. C. Elman, D. J. Silvester, and A. J. Wathen, *Finite Elements and Fast Iterative Solvers: With Applications in Incompressible Fluid Dynamics*, Oxford University Press, New York, 2005.
[12] H. C. Elman, D. J. Silvester, and A. J. Wathen, *Performance and analysis of saddle point preconditioners for the discrete steady-state Navier-Stokes equations*, Numer. Math., 90 (2002), pp. 665–688.
[13] M. A. Freitag and A. Spence, *Convergence theory for inexact inverse iteration applied to the generalised nonsymmetric eigenproblem*, Electron. Trans. Numer. Anal., 28 (2007), pp. 40–64.
[14] M. A. Freitag and A. Spence, *Convergence rates for inexact inverse iteration with application to preconditioned iterative solves*, BIT, 47 (2007), pp. 27–44.
[15] M. A. Freitag and A. Spence, *A tuned preconditioner for inexact inverse iteration applied to Hermitian eigenvalue problems*, IMA J. Numer. Anal., 28 (2007), pp. 522–551.
[16] M. A. Freitag and A. Spence, *Rayleigh quotient iteration and simplified Jacobi-Davidson method with preconditioned iterative solves*, Linear Algebra Appl., 428 (2008), pp. 2049–2060.
[17] M. A. Freitag and A. Spence, *Shift-invert Arnoldi's method with preconditioned iterative solves*, SIAM J. Matrix Anal. Appl., 31 (2009), pp. 942–969.
[18] L. Giraud, D. Ruiz, and A. Touhami, *A comparative study of iterative solvers exploiting spectral information for SPD systems*, SIAM J. Sci. Comput., 27 (2006), pp. 1760–1786.
[19] G. H. Golub and C. F. van Loan, *Matrix Computations*, 3rd ed., The Johns Hopkins University Press, Baltimore, MD, 1996.
[20] G. H. Golub, Z. Zhang, and H. Zha, *Large sparse symmetric eigenvalue problems with homogeneous linear constraints: The Lanczos process with inner-outer iterations*, Linear Algebra Appl., 309 (2000), pp. 289–306.
[21] C. R. Lee, *Residual Arnoldi Methods: Theory, Package and Experiments*, Ph.D. thesis, Department of Computer Science, University of Maryland, College Park, MD, 2007.
[22] R. B. Lehoucq, D. C. Sorensen, and C. Yang, *ARPACK Users' Guide: Solution of Large Scale Eigenvalue Problems by Implicitly Restarted Arnoldi Methods*. SIAM, Philadelphia, 1998.
[23] *Matrix Market*, http://math.nist.gov/MatrixMarket/.
[24] K. Meerbergen, A. Spence, and D. Roose, *Shift-invert and Cayley transforms for detection of rightmost eigenvalues of nonsymmetric matrices*, BIT, 34 (1994), pp. 409–423.
[25] R. B. Morgan, *GMRES with deflated restarting*, SIAM J. Sci. Comput., 24 (2002), pp. 20–37.
[26] M. L. Parks, E. De Sturler, G. Mackey, D. D. Johnson, and S. Maiti, *Recycling Krylov subspaces for sequences of linear systems*, SIAM J. Sci. Comput., 28 (2006), pp. 1651–1674.
[27] M. Robbé, M. Sadkane, and A. Spence, *Inexact inverse subspace iteration with preconditioning applied to non-Hermitian eigenvalue problems*, SIAM J. Matrix Anal. Appl., 31 (2009), pp. 92–113.
[28] V. Simoncini, *Variable accuracy of matrix-vector products in projection methods for eigencomputation*, SIAM J. Numer. Anal., 43 (2005), pp. 1155–1174.
[29] V. Simoncini and L. Eldén, *Inexact Rayleigh quotient-type methods for eigenvalue computations*, BIT, 42 (2002), pp. 159–182.

[30] V. Simoncini and D. B. Szyld, *Theory of inexact Krylov subspace methods and applications to scientific computing*, SIAM J. Sci. Comput., 25 (2003), pp. 454–477.
[31] V. Simoncini and D. B. Szyld, *Relaxed Krylov subspace approximation*, Proceedings in Applied Mathematics and Mechanics, 5 (2005), pp. 797–800.
[32] V. Simoncini and D. B. Szyld, *Recent computational developments in Krylov subspace methods for linear systems*, Numer. Linear Algebra Appl., 14 (2007), pp. 1–59.
[33] G. L. G. Sleijpen and J. van den Eshof, *On the use of harmonic Ritz pairs in approximating internal eigenpairs*, Linear Algebra Appl., 358 (2003), pp. 115–137.
[34] D. C. Sorensen, *Implicit application of polynomial filters in a k-step Arnoldi method*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 357–385.
[35] A. Stathopoulos, Y. Saad, and K. Wu, *Dynamic thick restarting of the Davidson, and the implicitly restarted Arnoldi methods*, SIAM J. Sci. Comput., 19 (1998), pp. 227–245.
[36] G. W. Stewart, *Matrix Algorithms. Vol.* II: *Eigensystems*, SIAM, Philadelphia, 2001.
[37] G. W. Stewart, *A generalization of Saad's theorem on Rayleigh-Ritz approximations*, Linear Algebra Appl., 327 (2001), pp. 115–119.
[38] G. W. Stewart, *An Unreliable Convergence Criterion for Arnoldi's Method*, Technical report, CMSC TR-4938, University of Maryland, College Park, MD, 2009.
[39] J. M. Tang, S. P. MacLachlan, R. Nabben, and C. Vuik, *A comparison of two-level preconditioners based on multigrid and deflation*, SIAM J. Matrix Anal. Appl., 31 (2010), pp. 1715–1739.
[40] J. van den Eshof and G. L. G. Sleijpen, *Inexact Krylov subspace methods for linear systems*, SIAM J. Matrix Anal. Appl., 26 (2005), pp. 125–153.
[41] J. van den Eshof, G. L. G. Sleijpen, and M. B. van Gijzen, *Relaxation strategies for nested Krylov methods*, J. Comput. Appl. Math., 177 (2005), pp. 347–365.
[42] F. Xue and H. C. Elman, *Fast inexact subspace iteration for generalized eigenvalue problems with spectral transformation*, Linear Algebra Appl., 435 (2011), pp. 601–622.