Harmonic and Refined Harmonic Shift-Invert Residual Arnoldi and Jacobi–Davidson Methods for Interior Eigenvalue Problems

Zhongxiao Jia† Cen Li‡

Abstract

This paper concerns the harmonic shift-invert residual Arnoldi (HSIRA) and Jacobi–Davidson (HJD) methods as well as their refined variants RHSIRA and RHJD for the interior eigenvalue problem. Each method needs to solve an inner linear system to expand the subspace successively. When the linear systems are solved only approximately, we are led to the inexact methods. We prove that the inexact HSIRA, RHSIRA, HJD and RHJD methods mimic their exact counterparts well when the inner linear systems are solved with only low or modest accuracy. We show that (i) the exact HSIRA and HJD expand subspaces better than the exact SIRA and JD and (ii) the exact RHSIRA and RHJD expand subspaces better than the exact HSIRA and HJD. Based on the theory, we design stopping criteria for inner solves. To be practical, we present restarted HSIRA, HJD, RHSIRA and RHJD algorithms. Numerical results demonstrate that these algorithms are much more efficient than the restarted standard SIRA and JD algorithms and furthermore the refined harmonic algorithms outperform the harmonic ones very substantially.

Keywords. Subspace expansion, expansion vector, inexact, low or modest accuracy, the SIRA method, the JD method, harmonic, refined, inner iteration, outer iteration.

AMS subject classifications. 65F15, 65F10, 15A18

1 Introduction

Consider the linear eigenproblem

$$Ax = \lambda x, \quad x^H x = 1,$$

where $A \in \mathbb{C}^{n \times n}$ is large and possibly sparse with the eigenvalues labeled as

$$0 < |\lambda_1 - \sigma| < |\lambda_2 - \sigma| \leq \cdots \leq |\lambda_n - \sigma|,$$

where $\sigma \in \mathbb{C}$ is a given target inside the spectrum of $A$. We are interested in the eigenvalue $\lambda_1$ closest to the target $\sigma$ and/or the associated eigenvector $x_1$. This is called the interior eigenvalue problem. We denote $(\lambda_1, x_1)$ by $(\lambda, x)$ for simplicity.

The interior eigenvalue problem arises from many applications [1]. Projection methods have been widely used for it [2][19][20][25][26]. For a given subspace $\mathcal{V}$, the standard projection
method, i.e., the Rayleigh–Ritz method, seeks the approximate eigenpairs, i.e., the Ritz pairs, $(\nu, y)$ satisfying
\[(A - \nu I)y \perp V, \quad y \in V. \quad (2)\]

As is well known, the standard projection method is quite effective for computing exterior eigenpairs, but is not for computing interior ones. For interior eigenpairs, one usually faces two difficulties when using the standard projection method: (i) the Ritz vectors might be very poor [7, 15] even if the subspace that contains good enough information on the desired eigenvectors; (ii) even though there are Ritz values which are good approximations to the desired eigenvalues, it is usually hard to recognize them and to select the correct Ritz pairs to approximate the desired interior eigenpairs [17, 18]. Because of possible mismatches, the Ritz pairs may misconverge or converge irregularly as subspace $V$ is expanded or restarted.

To overcome the above second difficulty, one popular approach is to apply the standard projection method to the shift-invert matrix $(A - \sigma I)^{-1}$, which transform the desired eigenvalues of $A$ into the dominant (exterior) ones. However, if it is very or too costly to factor $A - \sigma I$, as is often the case in practice, one must resort iterative solvers to approximately solve the linear systems involving $A - \sigma I$. Over years, one has focused the inexact shift-invert Arnold (SIA) type methods where inner linear systems are solved approximately. It has turned out that inner linear systems must be solved with very high accuracy when approximate eigenpairs are of poor accuracy [5, 21, 22]. As a result, it may be very expensive and even impractical to implement SIA type methods.

As an alternative, one uses the harmonic projection method, i.e., the harmonic Rayleigh–Ritz method, to seek the approximate eigenpairs, the harmonic Ritz pairs, $(\nu, y)$ satisfying
\[(A - (\nu I)y \perp (A - \sigma I)V, \quad y \in V. \quad (3)\]

The harmonic projection method has been used widely for the interior eigenvalue problem [2, 20, 25, 26]. The method can be derived by using the standard Rayleigh–Ritz method on $(A - \sigma I)^{-1}$ with respect to the subspace $(A - \sigma I)V$ without factoring $A - \sigma I$. The harmonic projection method has the advantage that \(\frac{1}{\mu - \sigma}\) is the Ritz value of $(A - \sigma I)^{-1}$. This is helpful to select the correct approximate eigenvalues since the desired interior eigenvalues near $\sigma$ have been transformed into the exterior ones that are much easier to match. So the harmonic projection method overcomes the second difficulty well.

However, the harmonic projection method inherits the first difficulty since the harmonic Ritz vectors may still be poor, converge irregularly and even fail to converge; see [12] and also [25, 26] for a systematical account. To this end, the refined projection method, i.e., the refined Rayleigh–Ritz method, has been proposed in [8, 15] that cures the possible non-convergence of the standard and harmonic Ritz vectors. The refined method replaces the Ritz vectors by the refined Ritz vectors that minimize residuals formed with the available approximate eigenvalues over the same projection subspace. The refined projection principle has also been combined with the harmonic projection method. Particularly, the harmonic and refined harmonic Arnoldi methods were proposed in [10] that are shown to be more effective for computing interior eigenpairs than the standard and refined Arnoldi methods. Due to the optimality, refined (harmonic) Ritz vectors are better approximations than (harmonic) Ritz vectors. The convergence of the refined projection methods has been established in a general setting [12, 15]; see also [25, 26].

A necessary condition for the convergence of any kind of projection methods is that the subspace contains enough accurate approximations to the desired eigenvectors. Therefore, a main ingredient for the success of a projection method is to construct a sequence of subspaces that contain increasingly accurate approximations to the desired eigenvectors. To achieve this
goal, it is naturally hoped that each expansion vector makes contribution to the desired
eigenvectors as much as possible. It turns out that preconditioning plays a key role in
obtaining good expansion vectors \([18, 23]\). Such kind of methods includes the Davidson
method \([3, 4]\), the Jacobi–Davidson (JD) method \([23]\) and shift-invert residual Arnoldi (SIRA)
method \([16]\).

The Davidson method can be seen as a preconditioning Arnoldi (Lanczos) method and a
prototype of the SIRA and JD methods, in which an inner linear system involving \(A - \sigma I\)
needs to be solved at each outer iteration. When a factorization of \(A - \sigma I\) is impractical, only
iterative solvers are viable for solving the inner linear systems. This leads to the inexact SIRA
and JD. A fundamental issue on them is how accurately we should solve inner linear systems
in order to make the inexact methods mimic the exact counterparts well, that is, the inexact
and exact ones use comparable outer iterations to achieve the convergence. For the simplified
or single vector JD method without subspace acceleration, it has been argued that it may not
be necessary to solve inner linear system with high accuracy and a low or modest accuracy
may suffice \([6, 23]\). This issue is more difficult to handle for the standard JD method. Let the
relative error of approximate solution of an inner linear system be \(\varepsilon\). Lee and Stewart \([16]\)
have given some analysis of the inexact SIRA under a series of assumptions. However, their
main results appear hard to justify or interpret, and furthermore no quantitative estimates
for \(\varepsilon\) have been given eventually. Using a new analysis approach, the authors \([14]\) have
established a rigorous general convergence theory of the inexact SIRA and JD in a unified
way and derived quantitative and explicit estimates for \(\varepsilon\). The results prove that a low or
moderate \(\varepsilon\), say \([10^{-4}, 10^{-3}]\), is generally enough to make the inexact SIRA and JD behave
very like the exact methods. As a result, the methods are expected to be more practical than
the inexact SIA method. The authors have confirmed the theory numerically in \([14]\).

Because of the merits of harmonic and refined harmonic projection methods for the interior
eigenvalue problem, in this paper we consider the harmonic and refined harmonic variants,
HSIRA, RHSIRA and HJD, RHJD, of the standard SIRA and JD methods. Exploiting
the analysis approach and some results in \([14]\), we establish the general convergence theory and
derive the estimates for \(\varepsilon\) for the four inexact methods. The results are similar to those for
the SIRA and JD methods in \([14]\). They prove that, in order to make the inexact methods
behave like their exact counterparts, it is generally enough to solve all the inner linear systems
with only low or modest accuracy. Furthermore, we show that the exact HSIRA and HJD
expand subspaces better than the exact SIRA and JD and, in turn, the exact RHSIRA
and RHJD expand subspaces better than the exact HSIRA and HJD. These results and
the merits of the harmonic and refined harmonic methods mean that the harmonic Ritz
vectors and refined harmonic Ritz vectors are better for subspace expansions and restarting
than the standard Ritz vectors. Based on the theory, we design stopping criteria for inner
solves. To be practical, we present restarted HSIRA, RHSIRA, HJD and RHJD algorithms.
We make numerical experiments to confirm our theory and demonstrate that the restarted
HSIRA, HJD and RHSIRA and RHJD algorithms are considerably more efficient than the
restarted standard SIRA and JD algorithms and furthermore the restarted RHSIRA and
RHJD outperform the others very greatly for the interior eigenvalue problem.

This paper is organized as follows. In Section \(2\) we describe the harmonic and refined
harmonic SIRA and JD methods. In Section \(3\) we present our results on \(\varepsilon\). In Section \(4\)
we describe restarted algorithms. In Section \(5\) we report numerical experiments to confirm our
theory. Finally, we conclude the paper and point out future work in Section \(6\).

Some notations to be used are introduced. Throughout the paper, denote by \(\| \cdot \|\)
the Euclidean norm of a vector and the spectral norm of a matrix, by \(I\) the identity matrix with
the order clear from the context, and by the superscript \(H\) the complex conjugate transpose
of a vector or matrix. We measure the deviation of a nonzero vector \( y \) from a subspace \( \mathcal{V} \) by
\[
\sin \angle(\mathcal{V}, y) = \frac{\| (I - P_{\mathcal{V}}) y \|}{\| y \|},
\]
where \( P_{\mathcal{V}} \) is the orthogonal projector onto \( \mathcal{V} \).

2 The harmonic SIRA and JD methods and their refined versions

Let the columns of \( V \) form an orthonormal basis of a given general subspace \( \mathcal{V} \) and define the matrices
\[
H = V^H (A - \sigma I)^H V, \tag{4}
\]
\[
G = V^H (A - \sigma I)^H (A - \sigma I) V. \tag{5}
\]
Then the standard projection method computes the Ritz pairs \((\nu, y)\) satisfying
\[
\begin{cases}
H^H z = \mu z \text{ with } \|z\| = 1, \\
\nu = \mu + \sigma, \\
y = V z,
\end{cases} \tag{6}
\]
and selects \( \nu \)'s closest to \( \sigma \), which correspond to the smallest \( \mu \)'s in magnitude, and the associated \( y \)'s to approximate the desired eigenpairs of \( A \). Given an initial subspace \( \mathcal{V} \) and the target \( \sigma \), we describe the standard SIRA and JD methods as Algorithm 1 for our later use.

**Algorithm 1** The SIRA/Jacobi–Davidson method with the target \( \sigma \)

Given the target \( \sigma \) and a user-prescribed convergence tolerance \( tol \), suppose the columns of \( V \) form an orthonormal basis of the initial subspace \( \mathcal{V} \).

repeat
1. Use the standard projection method to compute the Ritz pair \((\nu, y)\) of \( A \) with respect to \( \mathcal{V} \), where \( \nu \approx \lambda \) and \( \| y \| = 1 \).
2. Compute the residual \( r = Ay - \nu y \).
3. In SIRA, solve the linear system\[
(A - \sigma I) u_S = r. \tag{7}
\]
In JD, solve the correction equation\[
(I - yy^H)(A - \sigma I)(I - yy^H) u_J = -r \tag{8}
\]
for \( u_J \perp y \).
4. Orthogonalize the \( u_S \) or \( u_J \) against \( V \) to get the expansion vector \( v \).
5. Subspace expansion: \( V = \begin{bmatrix} V & v \end{bmatrix} \).
until \( \| r \| < tol \)

The harmonic projection method seeks the harmonic Ritz pairs \((\mu, y)\) satisfying (3), which amounts to solving the generalized eigenvalue problem of the pencil \((H, G)\):
\[
\begin{cases}
H z = \frac{1}{\mu} G z \text{ with } \|z\| = 1, \\
\nu = \mu + \sigma, \\
y = V z,\tag{9}
\end{cases}
\]
and selects $\nu$'s closest to $\sigma$, which corresponds to the smallest $\mu$'s in magnitude, and the associated $y$'s to approximate the desired eigenpairs of $A$.

Alternatively, we may use the Rayleigh quotient $\rho$ of $A$ with respect to the harmonic Ritz vector $y$ to approximate $\lambda$ as well. Recall the definition (4) of $H$ and note from (9) that

$$y = Vz$$

and

$$\|z\| = 1.$$ 

Then

$$\rho = \frac{y^H Ay}{y^H y} = \frac{z^H V^H (A - \sigma I + \sigma I) V z}{z^H V^H V z} = z^H H H z + \sigma,$$

so we can compute $\rho$ very cheaply. It is proved in [12] that once $\sigma$ is very close to the desired eigenvalue $\lambda$, then the harmonic projection method may miss $\lambda$ and the harmonic Ritz value $\mu$ may misconverge to some other eigenvalue; see [12, Theorem 4.1]. However, whenever $y$ converges to the desired eigenvector $x$, the Rayleigh quotient $\rho$ must converge to $\lambda$ no matter how close $\sigma$ is to $\lambda$; see [12, Theorem 4.2]. Therefore, it is better and safer to use $\rho$, rather than $\mu$, as an approximate eigenvalue. Another merit is that $\rho$ is optimal in the sense of

$$\rho = \arg \min_{\nu \in \mathbb{C}} \|(A - \nu I)y\|.$$ 

In the sequel, as was done in the literature, e.g., [12, 18], we will always use $\rho$ as the approximation to $\lambda$ in the harmonic projection method.

Define $r = (A - \rho I)y$. Then $(A - \rho I)y \perp y$. We replace $r$, the residual of standard Ritz pair, in (7) and (8) by this new $r$ and the standard Ritz vector $y$ in (8) by the harmonic Ritz vector. We then solve the resulting (7) and (8) with the new $r$ and $y$, respectively. With the solutions, we expand the subspaces analogous to Steps 4–5 of Algorithm 1. In such a way, we get the HSIRA and HJD methods.

We should point out that a harmonic JD method had already been proposed as early as the standard one in [23], where it is suggested to solve the following correction equation for $u'_{J} \perp y'$ (suppose that $y'Hy \neq 0$)

$$
\left( I - \frac{yy'H}{y'H y} \right) (A - \sigma I) \left( I - \frac{yy'H}{y'H y} \right) u'_{J} = -r
$$

with $r = (A - \mu I)y$ and $y' = (A - \sigma I)y$. It is different from (8) in the harmonic JD method described above, so its solution is different from that of (8) too. We prefer (8) since there is a potential danger that the oblique projector involved in (11) may make it worse conditioned than (8) considerably.

In what follows we propose the RHSIRA and RHJD methods. We first compute the Rayleigh quotient $\rho$ defined by (10) and then seek a unit length vector $\hat{y} \in \mathcal{V}$ satisfying the optimality property

$$\|(A - \rho I)\hat{y}\| = \min_{w \in \mathcal{V}, \|w\| = 1} \|(A - \rho I)w\|$$

and use it to approximate $x$. So $\hat{y}$ is the best approximation to $x$ from $\mathcal{V}$ with respect to $\rho$ and the Euclidean norm. We call $\hat{y}$ a refined harmonic Ritz vector or more generally a refined approximate eigenvector since the pair $(\rho, \hat{y})$ does not satisfy the harmonic projection (3) any more. Since $V$ forms an orthonormal basis of $\mathcal{V}$, (12) is equivalent to finding the unit length $\hat{z} \in \mathcal{C}^m$ such that $\hat{y} = V\hat{z}$ with $\hat{z}$ satisfying

$$\|(A - \rho I)\hat{y}\| = \min_{z \in \mathcal{C}^m, \|z\| = 1} \|(A - \rho I)V z\|$$

$$= \|(A - \rho I)V \hat{z}\|$$

$$= \sigma_{\min} ((A - \rho I)V).$$
We see that \( \hat{z} \) is the right singular vector associated with smallest singular value of the matrix \((A - \rho I)V\), and \( \hat{y} = V\hat{z} \). So we can get \( \hat{z} \) and \( \|(A - \rho I)V\hat{z}\| \) by computing the singular value decomposition (SVD) of \((A - \rho I)V\).

Similar to (10), define
\[
\hat{\rho} = \hat{y}^H A\hat{y} = \hat{z}H \hat{H} \hat{z} + \sigma, \tag{13}
\]
the Rayleigh quotient of \( A \) with respect to the refined harmonic Ritz vector \( \hat{y} \). Then the new residual \( \hat{r} = (A - \hat{\rho})\hat{y} \perp \hat{y} \). Replace \( r \) and \( y \) by \( \hat{r} \) and \( \hat{y} \) in (7) and (8) and perform Steps 3–5. Then we obtain the RHSIRA and RHIJD methods, respectively. We will use (\( \hat{\rho}, \hat{y} \)) to approximate \((\lambda, x)\) in the methods as \( \|(A - \hat{\rho}I)\hat{y}\| \leq \|(A - \rho I)\hat{y}\| \).

Some important results have been established in [11, 12] for standard and refined projection methods. The following two results in [11] are directly adapted to the harmonic and refined harmonic projection methods: First, we have \( \|(A - \rho I)\hat{y}\| < \|(A - \rho I)y\| \) unless the latter is zero, that is, the pair \((\rho, y)\) is an exact eigenpair of \( A \). Second, if \( \|(A - \rho I)y\| \neq 0 \) and there is another harmonic Ritz value close to \( \rho \), then it may occur that
\[
\|(A - \rho I)\hat{y}\| \ll \|(A - \rho I)y\|.
\]
So \( \hat{y} \) can be much more accurate than \( y \) as an approximation to the desired eigenvector \( x \).

For a general \( m \)-dimensional subspace \( \mathcal{V} \), two approaches are proposed in [9] for computing \( \hat{z} \). Approach I is to directly form the cross-product matrix
\[
S = V^H (A - \rho I)^H (A - \rho I)V \tag{14}
\]
which is Hermitian semi-positive definite. The desired \( \hat{z} \) is the normalized eigenvector associated with the smallest eigenvalue of \( S \), and \( \|(A - \rho I)\hat{y}\| \) is the square root of the smallest eigenvalue. Noticing that \( G \) and \( H \) are already available in the procedure of the harmonic projection, we can form \( S \) using at most \( 12m^2 \) flops by taking the worst case into account that all terms in (15) are complex. So, as a whole, we can compute \( \hat{z} \) by the QR algorithm at cost of \( O(m^3) + 12m^2 \) flops.

Approach II is to first make the thin or compact QR decomposition \( (A - \rho I)V = QR \) and then make the SVD of the \( m \times m \) triangular matrix \( R \) by \( O(m^3) \) flops to get the smallest singular value and the associated right singular vector, which are \( \|(A - \rho I)\hat{y}\| \) and \( \hat{z} \), respectively. If we use the Gram–Schmidt orthogonalization procedure with iterative refinement to compute the QR decomposition, then we will get a numerically orthonormal \( Q \) within a small multiple of the machine precision, which totally needs no more at most \( 4nm^2 \) flops generally if \( \rho \) is real. As a whole, \( 4nm^2 + O(m^3) \) flops are needed.

By comparison, we find that Approach I is, computationally, much more effective than Approach II. It can be justified [13] that Approach I is numerically stable for computing \( \hat{z} \) provided that there is a considerable gap between the smallest singular value and the second smallest one of \((A - \rho I)V\). So in this paper, we use Approach I to compute \( \hat{y} \).

Because of different right-hand sides, it is important to note that expanded subspaces are generally different for the SIRA, HSIRA and RSIRA methods whatever the linear systems are solved either exactly or approximately. The same is true for the JD, HJD and RHIJD methods because of not only different right-hand sides but also different \( y \) in the correction equation [5]. This is different from shift-invert Arnoldi (SIA) type methods, i.e., the standard SIA, harmonic SIA, refined SIA and refined harmonic SIA, where the the updated subspaces are the same once the linear systems are solved exactly or approximately using the same inner solver with the same accuracy because the right-hand sides involved are always the currently newest basis vector at each outer iteration step. We will come back to this point in the end of Section [8] and show which method favors subspace expansion.
3 Accuracy requirements of inner iterations in HSIRA, HJD, RHSIRA and RHJD

In this section, we review some important results in [14] and apply them to establish the convergence theory of the inexact HSIRA, HJD and RHSIRA and RHJD. We prove that each inexact method mimics its exact counterpart well provided that all the inner linear systems are solved with only low or modest accuracy. We stress that in the presentation $y$ is a general approximation to $x$. Returning to our methods, $y$ is just the harmonic Ritz vector $y$ in HSIRA and HJD and the refined harmonic Ritz vector $\hat{y}$ in RHSIRA and RHJD. Finally, we look into the way that each exact method expands the subspace and make a simple analysis, showing that HSIRA, HJD and RHSIRA, RHJD generally expand the subspace more effectively than SIRA and JD when computing interior eigenpairs, so that the former ones are more effective than the latter ones. This advantage conveys to the inexact HSIRA, HJD and RHSIRA, RHJD when each inexact method mimics its exact counterpart well.

We can write the linear system (7) and the correction equation (8) as a unified form

$$ (A - \sigma I)u = \alpha_1 y + \alpha_2 (A - \sigma I)y, $$

which leads to (7) in the HSIRA or RHSIRA method and (8) in the HJD or RHJD method when $\alpha_1 = \sigma - \nu$, $\alpha_2 = 1$ or $\alpha_1 = \sigma - \rho$, $\alpha_2 = 1$ and $\alpha_1 = \frac{1}{y^H By}$, $\alpha_2 = 1$, respectively. Define $B = (A - \sigma I)^{-1}$. Then the exact solution $u$ of (16) is

$$ u = \alpha_1 By + \alpha_2 y. $$

Let $\tilde{u}$ be an approximate solution of (16) and its relative error be

$$ \varepsilon = \frac{\|\tilde{u} - u\|}{\|u\|}. $$

Then we can write

$$ \tilde{u} = u + \varepsilon\|u\|f, $$

where $f$ is the normalized error direction vector. Note that the direction of $f$ depends on inner linear systems solves and accuracy requirements, but it shows no particular features in general. So $\sin \angle(V, f)$ is generally moderate and not near zero. The vectors $u$ and $\tilde{u}$ are orthogonalized against $V$ to get the (unnormalized) expansion vectors $(I - P_V)u$ and $(I - P_V)\tilde{u}$, respectively, where $P_V$ is the orthogonal projector onto $V$. Define the relative error

$$ \tilde{\varepsilon} = \frac{\|((I - P_V)\tilde{u} - (I - P_V)u)\|}{\|(I - P_V)u\|}, $$

which measures the relative difference between two expansion vectors $(I - P_V)u$ and $(I - P_V)\tilde{u}$. The following result [14, Theorem 3] establishes a compact bound for $\varepsilon$ in terms of $\tilde{\varepsilon}$.

**Theorem 1.** Define $B = (A - \sigma I)^{-1}$ and $\alpha = \frac{-\alpha_2}{\alpha_1}$ with $\alpha_1 \neq 0$, where $\alpha_1$ and $\alpha_2$ are given in (16). Suppose $\left(\frac{1}{\lambda - \sigma}, x\right)$ is a simple desired eigenpair of $B$ and let $(x, X_\perp)$ be unitary. Then

$$ \left[ \begin{array}{c} x^H \\ X_\perp^H \end{array} \right] B \left[ \begin{array}{c} x \\ X_\perp \end{array} \right] = \left[ \begin{array}{c} \frac{1}{\lambda - \sigma} \\ 0 \end{array} \right] c^H , $$

where $c^H = x^H BX_\perp$ and $L = X_\perp^H BX_\perp$. Assume that $\sin \angle(V, f) \neq 0$ and $\alpha$ is not an eigenvalue of $L$ and define

$$ \text{sep} (\alpha, L) = \|L^{-1}\|^{-1} > 0. $$
Then
\[ \varepsilon \leq C \tilde{\varepsilon}, \]
where
\[ C := \frac{2\|B\|}{\text{sep}(\alpha, L) \sin \angle(V, f)} \]
with \( \alpha = \frac{1}{\nu - \sigma} \) or \( \alpha = \frac{1}{\rho - \sigma} \) in SIRA type methods and \( \alpha = y^H B y \) or \( \hat{y}^H B \hat{y} \) in JD type methods.

If \( \alpha \) is a fairly approximation to \( \frac{1}{\lambda - \sigma} \) and \( \frac{1}{\lambda - \sigma} \) is not close to any eigenvalue of \( L \), then \( \text{sep}(\alpha, L) \) is not small and is actually of \( O(\|B\|) \) provided that the eigensystem of \( B \) is not ill conditioned. In this case, \( C \) is moderate as \( \sin \angle(V, f) \) is moderate, as commented previously.

For a given \( \tilde{\varepsilon} \), we should note that the bigger \( \sin \angle(V, f) \) is, the bigger \( \varepsilon \) is. So it is a lucky event if \( \sin \angle(V, f) \) is big as it means that we need to solve inner linear systems with less accuracy \( \varepsilon \).

Below we discuss how to determine \( \tilde{\varepsilon} \) such that the inexact methods mimic their exact counterparts very well, that is, each inexact method and its exact counterpart use comparable or almost the same outer iterations to achieve the convergence. The following important result is proved in [14], which forms the basis of our analysis.

**Theorem 2.** Define
\[ V_+ = V \oplus \text{span}\{(I - P_V)u\}, \quad \tilde{V}_+ = V \oplus \text{span}\{(I - P_V)\tilde{u}\}, \]
where \( u \) and \( \tilde{u} \) are the exact and inexact solutions of the linear system (16), respectively, and
\[ \delta = \frac{\sin \angle(V_+, x)}{\sin \angle(V, x)}, \quad \tilde{\delta} = \frac{\sin \angle(\tilde{V}_+, x)}{\sin \angle(V, x)}. \]

Then we have
\[ \left| \frac{\delta - \tilde{\delta}}{\delta} \right| \leq 2 \frac{\tilde{\varepsilon}}{\delta} := \tau. \] (24)

Furthermore, if \( \tau < 1 \), then
\[ \frac{\sin \angle(\tilde{V}_+, x)}{\sin \angle(V_+, x)} = \frac{\tilde{\delta}}{\delta} \in [1 - \tau, 1 + \tau]. \] (25)

(23) measures one step subspace improvements of the exact and inexact subspace expansions, respectively. (25) indicates that to make \( \sin \angle(V_+, x) \approx \sin \angle(V_+, x) \), \( \tau \) should be small. Note that the difference of the upper and lower bounds is 2\( \tau \). So a very small \( \tau \) only improves the bounds very marginally, and a fairly small \( \tau \), e.g., \( \tau \in [0.001, 0.01] \), is enough since we have \( \frac{\sin \angle(\tilde{V}_+, x)}{\sin \angle(V_+, x)} \in [0.999, 1.001] \) or \([0.99, 1.01]\) and the lower and upper bounds differ marginally, which means that the two subspaces \( \tilde{V}_+ \) and \( V_+ \) are of comparable or almost the same quality for a fairly small \( \tau \) when computing \( (\lambda, x) \). Precisely, with respect to the two subspaces the approximations of the desired \((\lambda, x)\) obtained by the same type, i.e., the Rayleigh–Ritz method, its harmonic variant or the refined harmonic variant, should generally have the comparable or almost the same accuracy. By (21), we have
\[ \tilde{\varepsilon} = \frac{\tau \times \delta}{2}. \] (26)

As it has been proved in [14], the size of \( \delta \) crucially depends on the eigenvalue distribution of \( B \). Generally speaking, the better \( \frac{1}{\lambda - \sigma} \) is separated from the other eigenvalues of \( B \), the
smaller $\delta$ is. Conversely, if $\frac{1}{\lambda - \sigma}$ is poorly separated from the others, $\delta$ may be near to one; see [14] for a descriptive analysis. $\delta$ is an a-priori quantity and is unknown during computation. But generally we should not expect that a practical problem is very well conditioned, that is, $\delta$ is not much less than one; otherwise, the exact methods will generally find $(\lambda, x)$ by using only a very few outer iterations. For example, suppose that the initial $V$ is one dimensional and $\delta = 0.1$ at each outer iteration. Then the updated $\sin \angle(V_+, x)$ is no more than $10^{-10}$ after ten outer iterations and $V_+$ is a very accurate subspace for computing $x$. So in practice, we assume that $\delta$ is not much less than one, say no smaller than $0.2$. As we have argued, a fairly small $\tau \in [0.001, 0.01]$ should make the inexact methods behave very like their exact counterparts. As a result, by (26) and the argument that a fairly small $\tau \in [0.001, 0.01]$ is generally enough, to make the inexact methods mimic their exact counterparts, it is reasonable to take

$$\tilde{\varepsilon} \in [10^{-4}, 10^{-3}],$$

which is also suggested in [14] for the standard SIRA and JD methods. For $\tau \in [0.001, 0.01]$, if $\tilde{\varepsilon}$ defined by (27) is unfortunately bigger than that in (26), the the inexact methods may use more outer iterations and may not behave very like the exact methods. Even so, however, since solving inner linear systems with $\tilde{\varepsilon}$ in (27) is cheaper than with considerably smaller $\tilde{\varepsilon}$, the overall efficiency of the inexact methods may still be improved.

Finally, for the exact methods, let us, qualitatively and concisely, show which updated subspace $V_+$ is better and which method behaves more favorable. Keep (17) and $y \in V$ in mind. The (unnormalized) expansion vector is

$$(I - P_V)u = (I - P_V)By.$$

So we actually use add $By$ to $V$ to get the expanded subspace $V_+$. It is easily justified that the better $y$ approximates $x$, the better does $By$ in direction. So for a more accurate approximate eigenvector $y$, $By$ and thus the expanded subspace $V_+$ contain richer information on the eigenevector $x$. As we have argued, since the harmonic Ritz vector is a more reliable and regular approximation to $x$ while the Ritz vector may not be for the interior eigenvalue problem, so the exact HSIRA and HJD may expand the subspaces better and more regularly than the exact SIRA and JD. Furthermore, since the refined harmonic Ritz vector is generally more and can be much more accurate than the harmonic Ritz vector, the exact RHSIRA and RHJD, in turn, generate better subspaces than the exact HSIRA and HJD at each outer iteration. As a consequence, HSIRA and HJD are expected to converge more regularly and use fewer outer iterations than SIRA and JD while RHSIRA and RHJD may use the fewest outer iterations among all the exact methods for the interior eigenvalue problem. For the inexact methods, provided that the selection of $\tilde{\varepsilon}$ makes each inexact method mimic its exact counterpart well, the inexact HSIRA, HJD and RHSIRA, RHJD are advantageous to the inexact SIRA and JD. Numerical experiments will confirm our expectations.

4 Restarted algorithms and stopping criteria for inner solves

Due to the storage requirement and computational cost, it is generally necessary to restart the methods to avoid large steps of outer iterations. We describe the restarted HSIRA/HJD algorithms as Algorithm 2 and their refined variants as Algorithm 3.

We consider some practical issues on two algorithms. For outer iteration steps $m = 1, 2, \ldots, M_{\text{max}}$ during the current cycle, suppose $(y^{(m)}, y^{(m)})$ is used to approximate the desired eigenpair $(\lambda, x)$ of $A$ at the $m$-th outer iteration. As done commonly in the literature,
Algorithm 2 Restarted HSIRA/HJD algorithm with the fixed target $\sigma$

Given the target $\sigma$ and a user-prescribed convergence tolerance $tol$, suppose the columns of $V$ form an orthonormal basis of an initial subspace $\mathcal{V}$ and let $M_{\text{max}}$ be the maximum of outer iterations allowed.

repeat
1. Compute (update) $H = V^H (A - \sigma I)^H V$ and $G = V^H (A - \sigma I)^H (A - \sigma I)V$.
2. Let $(\mu, z)$ be an eigenpair of the matrix pencil $(H, G)$, where $\mu \equiv \lambda$.
3. Compute the Rayleigh quotient $\rho = z^H H^H z + \sigma$ and the harmonic Ritz vector $y = Vz$.
4. Compute the residual $r = Ay - py$.
5. In HSIRA or HJD, solve the inner linear system $(A - \sigma I)u = r \quad \text{or} \quad (I - yy^H)(A - \sigma I)(I - yy^H)u = -r$.
6. Orthonormalize $u$ against $V$ to get the expansion vector $v$.
7. If $\dim(\mathcal{V}) < M_{\text{max}}$, set $V = [V \ v]$; otherwise, $V = V_{\text{new}}$.
until $\|r\| < tol$

Algorithm 3 Restarted RHSIRA/RHJD algorithm with the fixed target $\sigma$

Given the target $\sigma$ and a user-prescribed convergence tolerance $tol$, suppose the columns of $V$ form an orthonormal basis of an initial subspace $\mathcal{V}$ and let $M_{\text{max}}$ be the maximum of outer iterations allowed.

repeat
1. Compute (update) $H = V^H (A - \sigma I)^H V$ and $G = V^H (A - \sigma I)^H (A - \sigma I)V$.
2. Let $(\mu, z)$ be an eigenpair of the matrix pencil $(H, G)$, where $\mu \equiv \lambda$.
3. Compute the Rayleigh quotient $\rho = z^H H^H z + \sigma$.
4. Form the cross-product matrix $S = G + (\sigma - \rho)H + (\sigma - \rho)H + |\sigma - \rho|^2 I$ and compute the eigenvector $\hat{z}$ of $S$ associated with its smallest eigenvalue.
5. Compute the new Rayleigh quotient $\rho = \hat{z}^H H^H \hat{z} + \sigma$ and the refined harmonic Ritz vector $y = V\hat{z}$.
6. Compute the residual $r = Ay - py$.
7. In RHSIRA or RHJD, solve the inner linear system $(A - \sigma I)u = r \quad \text{or} \quad (I - yy^H)(A - \sigma I)(I - yy^H)u = -r$.
8. Orthonormalize $u$ against $V$ to get the expansion vector $v$.
9. If $\dim(\mathcal{V}) < M_{\text{max}}$, set $V = [V \ v]$; otherwise, $V = V_{\text{new}}$.
until $\|r\| < tol$

we simply take the new starting vector

$$V_{\text{new}} = y^{(M_{\text{max}})}$$

to restart the algorithms. In practice, if $y^{(M_{\text{max}})}$ is complex conjugate, we take their real and imaginary parts, normalize and orthonormalize them to get an orthonormal $V_{\text{new}}$ of column two. We mention that a thick restarting technique [24] may be used, in which, besides $y^{(M_{\text{max}})}$, one retains the approximate eigenvectors associated with a few other approximate eigenvalues closest to $\rho^{(M_{\text{max}})}$ and orthonormalizes them to obtain $V_{\text{new}}$. We will not report numerical results with thick restarting.
Note that all the methods under consideration do not have residual monotonically decreasing properties as outer iteration steps increase. Therefore, it may be possible for both Algorithm 2 and Algorithm 3 to take bad restarting vectors. This is indeed the case for the standard Rayleigh–Ritz method when computing interior eigenpairs, as has been widely realized, e.g., [26]. However, as was stated previously, the harmonic and refined harmonic methods are more reliable to select correct and good approximations to the desired eigenpairs. So they are more suitable and reliable for restarting than the standard Ritz vector for the interior eigenvalue problem. As a consequence, Algorithm 2 and especially Algorithm 3 converge more regularly than the restarted standard SIRA and JD. Our numerical examples will confirm this and illustrate that the refined harmonic Ritz vectors are better than the harmonic Ritz vectors for restarting and Algorithm 3 converges faster than Algorithm 2.

Our ultimate goal is to get a practical estimate for $\varepsilon$ for a given $\tilde{\varepsilon}$. The a-priori bound (21) forms the basis for this. Let $\nu_1, \nu_2, \ldots, \nu_m$ are the harmonic Ritz values, i.e., the eigenvalues of the pencil $(H_m, G_m)$ at the $m$-th outer iteration during the current cycle (here we add the subscript $m$ to $H$ and $G$ in the algorithms) and assume that $\nu_1$ is used to approximate the desired eigenvalue $\lambda$. We simply estimate $\|B\| \approx \rho - \sigma$, where $\rho$ is the Rayleigh quotient of $A$ with respect to the harmonic Ritz vector in HSIRA and HJD and the refined harmonic Ritz vector in RHSIRA and RHJD. For $\text{sep}(\alpha, L)$, we replace $\alpha$, an approximation to $\frac{1}{\lambda - \sigma}$, by $\frac{1}{\rho - \sigma}$ accordingly and then estimate

$$\text{sep}(\alpha, L) \approx \min_{i=2,3,\ldots,m} \left| \frac{1}{\rho - \sigma} - \frac{1}{\nu_i - \sigma} \right|.$$ 

Finally, replace $\sin \angle(V, f)$ by its maximum one. Combining all these together, we get the following estimate $C'$ of $C$ in (21):

$$C' = \left\{ \begin{array}{ll}
2 \max_{i=2,3,\ldots,m} \left| \frac{\nu_i - \sigma}{\nu_i - \rho} \right|, & m > 1 \\
1, & m = 1
\end{array} \right.,$$

(28)

This is analogous to what is done in [14], and the difference is that we here use the harmonic Ritz values $\nu_2, \ldots, \nu_m$ to replace the standard Ritz values used in [14] as approximations to some eigenvalues of $A$ other than $\lambda$. Denote by $\varepsilon_S$ and $\varepsilon_J$ practical $\varepsilon$’s used in the HSIRA, RHSIRA and HJD, RHJD algorithms, respectively. Recall that bound (21) is compact. Then we take

$$\varepsilon_S = \varepsilon_J = \varepsilon = C'\tilde{\varepsilon}.$$  

(29)

For a fairly small $\tilde{\varepsilon}$, we may have $\varepsilon \geq 1$ in case $C'$ is big, which will make $\tilde{u}$ no sense as an approximation to $u$. In order to make $\tilde{u}$ have some accuracy, we propose using

$$\varepsilon = \min \{C'\tilde{\varepsilon}, 0.1\},$$  

(30)

which shows that $\varepsilon$ is comparable to $\tilde{\varepsilon}$ in size whenever $\tilde{\varepsilon}$ is small and $C'$ is moderate.

5 Numerical experiments

All the numerical experiments were performed on an Intel (R) Core (TM)2 Quad CPU Q9400 2.66GHz with the main memory 2 GB using Matlab 7.8.0 with the machine precision $\epsilon_{\text{mach}} = 2.22 \times 10^{-16}$ under the Linux operating system. All the test examples are difficult in the sense that the desired eigenvalue is clustered with some other eigenvalues of $A$ for the given target $\sigma$. We aim to show four points. First, regarding restarts of outer iterations,
for fairly small $\tilde{\varepsilon} = 10^{-3}$ and $10^{-4}$, the restarted inexact HSIRA/HJD and RHSIRA/RHJD algorithms behave (very) like the exact counterparts. Second, regarding outer iterations, Algorithms 2–3 are much more efficient than the restarted standard SIRA and JD for the same $\tilde{\varepsilon}$’s, and Algorithm 3 is the best. Third, regarding total inner iterations and overall efficiency, Algorithm 2 is considerably more efficient than Algorithm 1 and the restarted inexact standard SIRA and JD perform very poorly and often fail to converge. Fourth, each SIRA type algorithm is equally as efficient as the corresponding JD type algorithm for the same $\tilde{\varepsilon}$.

At the $m$-th outer iteration step of the inexact HSIRA or HJD method, we have $H_m = V_m^H(A - \sigma I)^H V_m$ and $M_m = V_m^H(A - \sigma I)^H(A - \sigma I)V_m$. Let $(\nu_i^{(m)}; V_{m-i}^{(m)})$, $i = 1, 2, \ldots, m$ be the harmonic Ritz pairs, labeled as $|\nu_1^{(m)} - \sigma| < |\nu_2^{(m)} - \sigma| \leq \cdots \leq |\nu_m^{(m)} - \sigma|$. Keep (10) in mind. We use

$$ (\rho^{(m)}, y^{(m)}) = \left( (z_1^{(m)})^H H_m z_1^{(m)} + \sigma, V_m z_1^{(m)} \right) $$

(31)

to approximate the desired eigenpair $(\lambda, x)$ of $A$. If the RSIRA and RHJD methods are used, we form the cross-product matrix $S_m$ by (15) and compute its eigenvector $\hat{z}^{(m)}$ associated with the smallest eigenvalue. We then compute the refined harmonic Ritz vector $y^{(m)} = V_m \hat{z}^{(m)}$ and the Rayleigh quotient $\rho^{(m)}$ defined by (13). Let $r_m = A y^{(m)} - \rho^{(m)} y^{(m)}$. We stop the algorithms if

$$ \|r_m\| < tol = \max \{\|A\|_1, 1\} \times 10^{-12}. $$

(32)

In the algorithms, we stop inner iterations at outer iteration $m$ when

$$ \frac{\|r_m - (A - \sigma I) \tilde{u}\|}{\|r_m\|}, \quad \frac{\| - r_m - (I - y^{(m)} (y^{(m)})^H)(A - \sigma I)(I - y^{(m)} (y^{(m)})^H) \tilde{u}\|}{\|r_m\|} \leq \varepsilon, $$

(33)

where $\varepsilon$ is defined by (30) for a given $\tilde{\varepsilon}$. We will denote by SIRA($\tilde{\varepsilon}$), JD($\tilde{\varepsilon}$), HSIRA($\tilde{\varepsilon}$), HJD($\tilde{\varepsilon}$) and RHSIRA($\tilde{\varepsilon}$), RHJD($\tilde{\varepsilon}$) the inexact algorithms with the given $\tilde{\varepsilon}$.

In the “exact” SIRA and JD type algorithms, we stop inner iterations when (33) is satisfied with $\varepsilon = 10^{-14}$.

All the test problems are from Matrix Market [1]. For each inner linear system, we always took the zero vector as an initial approximate solution and solved it by the right-preconditioned restarted GMRES(30) algorithm. Each algorithm starts with the normalized vector $\frac{1}{\sqrt{n}}(1, 1, \ldots, 1)^H$. At the $m$-th outer iteration of current restart, $m = 1, 2, \ldots, M_{\text{max}}$, for the correction equations in the JD type algorithms we used the preconditioner

$$ M_m = \left( I - y^{(m)} (y^{(m)})^H \right) M \left( I - y^{(m)} (y^{(m)})^H \right)^H $$

(34)

suggested in [26], where $M \approx A - \sigma I$ is the incomplete LU preconditioner for the SIRA type algorithms. For all the algorithms, the maximum steps of outer iterations are 30 per restart. An algorithm signals the failure if it did not converge within $\text{Max}$ restarts.

In the experiments, each figure consists of three subfigures, in which the top subfigure denotes outer residual norms versus outer iterations of the first cycle before restart, the bottom-left subfigure denotes outer residual norms versus restarts and the bottom-right subfigure denotes the numbers of inner iterations versus restarts.

The top subfigures exhibit the qualities of the standard, harmonic and refined harmonic Ritz vectors for restarting. The bottom two subfigures depict the convergence processes of the exact SIRA type algorithms and their inexact counterparts with $\tilde{\varepsilon} = 10^{-3}$, showing the convergence behavior of the algorithms and the local efficiency per restart, respectively.
In all the table, denote by $I_{\text{restart}}$ the number of restarts to achieve the convergence, by $I_{\text{inner}}$ the total number of inner iterations and by $P_{0.1}$ the percentage of the times $I_{0.1}$ that $\varepsilon = 0.1$ is used in the total number of outer iterations $I_{\text{outer}}$, i.e.,

$$P_{0.1} = \frac{I_{0.1}}{I_{\text{outer}}}.$$  

Note that $I_{\text{inner}}$ equals the total products of $A$ and vectors in the restarted GMRES algorithm. It is a reasonable measure of the overall efficiency of all the algorithms used in the experiments.

**Example 1.** This unsymmetric eigenproblem $M80PL_n$ of $n = 4182$ arises from real power system models $[1]$. We test this example with $\sigma = 0.05 + 0.5i$ and $\text{Max} = 500$. The computed eigenvalue is $\lambda \approx -6.9391 \times 10^{-5} + 5.0062 \times 10^{-1}i$. The preconditioner $M$ is the incomplete LU factorization of $A - \sigma I$ with drop tolerance $10^{-1}$. Figure 1 and Table 1 display the results.

![Figure 1: Example 1. M80PL$_n$ with $\sigma = 0.05 + 0.5i$.](image)

We see from the top subfigure of Figure 1 that for $\tilde{\varepsilon} = 10^{-3}$ the inexact SIRA, HSIRA and RHSIRA behaved very like their corresponding exact counterparts in the first cycle but the exact and inexact RHSIRA are more effective than the exact and inexact HSIRA. It clearly shows that the exact SIRA and SIRA($10^{-3}$) are the poorest and considerably poorer than the corresponding HSIRA and RHSIRA. There are two reasons for this. The first is that harmonic and refined harmonic Ritz vectors are more reliable than standard Ritz vectors for computing interior eigenvectors. The second is that the harmonic and refined harmonic Ritz vectors favor subspace expansions. We, therefore, expect that the restarted HSIRA...
Accuracy | Algorithm | $I_{\text{restart}}$ | $I_{\text{inner}}$ | $P_{0.1}$ | Algorithm | $I_{\text{restart}}$ | $I_{\text{inner}}$ | $P_{0.1}$
---|---|---|---|---|---|---|---|---
$\tilde{\varepsilon} = 10^{-3}$ | SIRA | Max | 32142 | 0% | JD | Max | 33823 | 0%
 | HSIRA | 383 | 35032 | 0% | HJD | 371 | 33988 | 0%
 | RHSIRA | 79 | 4981 | 89% | RHJD | 79 | 4958 | 89%
$\tilde{\varepsilon} = 10^{-4}$ | SIRA | Max | 47166 | 0% | JD | Max | 47785 | 0%
 | HSIRA | 376 | 46470 | 0% | HJD | 374 | 46231 | 0%
 | RHSIRA | 74 | 7032 | 0% | RHJD | 75 | 7148 | 0%
exact | SIRA | Max | 217941 | – | JD | Max | 217936 | –
 | HSIRA | 368 | 171183 | – | HJD | 368 | 171182 | –
 | RHSIRA | 76 | 35515 | – | RHJD | 76 | 35513 | –

Table 1: Example 1. M80PI_n with $\sigma = 0.05 + 0.5i$.

and RHSIRA can be much more efficient and converge much faster than the restarted SIRA when computing interior eigenpairs since we use more reliable and possibly accurate harmonic and refined harmonic Ritz vectors as restarting vectors. Furthermore, as far as restarts are concerned, the restarted RHSIRA outperforms the restarted HSIRA very substantially. We observed very similar convergence behavior for the JD type algorithms and thus have the same expectations on the restarted JD type algorithms. These expectations are indeed confirmed by numerical experiments, as shown by $I_{\text{restart}}$’s in the table and figure.

We explain the table and figure in more details. From the bottom-left subfigure, regarding outer iterations, we see that both the exact and inexact restarted SIRA did not converge within 500 restarts while HSIRA and RHSIRA worked well and the inexact methods behaved very like their corresponding exact ones. The restarted RHSIRA and RHJD were the fastest and five times as fast as the restarted HSIRA and HJD, respectively. As is expected, the table confirms that, for the same $\tilde{\varepsilon}$, each SIRA type algorithm and the corresponding JD type algorithm behaved very similar and were almost indistinguishable.

Regarding the overall efficiency, the exact HSIRA, HJD, RHSIRA and RHJD each used about 460 inner iterations per restart. In contrast, HSIRA($10^{-3}$) and HSIRA($10^{-4}$) used almost constant inner iterations each restart, which were about 100 and 110 per restart, respectively, and RHSIRA($10^{-3}$) and RHSIRA($10^{-4}$) used roughly 60 and 100 inner iterations per restart, respectively. The same observations are true for the JD type algorithms. These experiments demonstrate that modest $\tilde{\varepsilon} = 10^{-3}$, $10^{-4}$ are enough to make the inexact restarted algorithms mimic their exact counterparts. We find that the SIRA($10^{-4}$) and JD($10^{-4}$) type algorithms used almost the same outer iterations as the SIRA($10^{-3}$) and JD($10^{-3}$) type ones, but the latter consumed considerably fewer total inner iterations and improved the overall efficiency substantially. So, smaller $\tilde{\varepsilon}$’s are not necessary for this example as they cannot reduce outer iterations further and may cost much more inner iterations.

In addition, we see from Table 1 that 89 percent of inner linear systems in RHSIRA($10^{-3}$) were actually solved with the accuracy requirement $\varepsilon = 0.1$. This means that though most of $C^\prime$s in (28) are big, it suffices to solve all the inner linear systems with low accuracy 0.1.

**Example 2.** This unsymmetric eigenproblem M80PL_n of $n = 4182$ arises from real power system models [1]. We test test this example with $\sigma = 0.4 + 1.3i$ and Max = 1000. The computed eigenvalue is $\lambda \approx -2.8266 \times 10^{-4} + 1.3068i$. The preconditioner $M$ is the incomplete LU factorization of $A - \sigma I$ with drop tolerance $10^{-1}$. Figure 2 and Table 2 report the results.

The top subfigure of Figure 2 is similar to that of Figure 1 showing that harmonic and especially refined harmonic Ritz vectors are more suitable for expanding subspaces and restarting for the interior eigenvalue problem. The difference is that this problem is more difficult than Example 1 since, for the 30-dimensional subspace in the first cycle, the residual norms decrease more slowly and approximate eigenpairs are less accurate than those for
Example 1. So it is expected that the restarted algorithms converge more slowly and use more outer iterations \( I_{\text{restart}} \)’s than for Example 1. We observe from the figure that the convergence curves of the three exact algorithms SIRA, HSIRA and RHSIRA essentially coincide with those of their inexact variants with \( \tilde{\varepsilon} = 10^{-3} \). So smaller \( \tilde{\varepsilon} \) does not help, rather it makes the algorithms may waste much more inner iterations. We observed similar convergence phenomena for the JD type algorithms.

Table 2 and Figure 2 shows that the restarted exact and inexact SIRA, JD, HSIRA and HJD all failed to converge within 1000 restarts. Furthermore, it is deduced from Figure 2 that the restarted exact and inexact SIRA and JD appear impossible to converge at all since their convergence curves were irregular, oscillated and had no decreasing tendency. The restarted exact and inexact SIRA and JD behaved regular but converged too slowly. In contrast,
the restarted exact and inexact RHSIRA and RHJD with $\tilde{\varepsilon} = 10^{-3}$ and $10^{-4}$ converged. Furthermore, the restarted RHSIRA and RHJD with $\tilde{\varepsilon} = 10^{-3}$ and $\tilde{\varepsilon} = 10^{-4}$ behaved very similar and used almost the same outer iterations. Both of them mimic the restarted exact RHSIRA and RHJD fairly well. So, smaller $\varepsilon$ cannot reduce outer iterations substantially.

Regarding the overall efficiency, the inexact restarted algorithms improved the situation tremendously. As in Example 1, the exact RHSIRA and RHJD still needed about 370 inner iterations per restart, much more than 40 inner iterations that were used by the inexact RHSIRA and RHJD for $\tilde{\varepsilon} = 10^{-3}$ and $10^{-4}$ per restart. As a whole, $I_{inner}$’s illustrate that the inexact algorithms with these two $\tilde{\varepsilon}$’s were about eight times faster than the exact algorithms, a striking improvement. Besides, note that the number of inner linear systems that were solved with lower accuracy $\varepsilon = 0.1$ in RHSIRA($10^{-3}$) were 96%, while those in HSIRA($10^{-3}$) and SIRA($10^{-3}$) were 0% and 6%, respectively. This is why RHSIRA($10^{-3}$) used fewer inner iterations than SIRA($10^{-3}$) and HSIRA($10^{-3}$) per restart, as shown in the bottom-right subfigure.

Although the restarted exact SIRA and HSIRA both failed for this example, the reasons may be completely different. In the top subfigure, we find that the convergence curves of SIRA bulged at the last few steps (25 $\sim$ 30). This is harmful to restarting as it is very possible to get an unsatisfying restarting vector once the method bulged at the very last step. In the bottom-left subfigure, it is seen that the convergence curves of restarted exact and inexact SIRA were irregular while the convergence curves of HSIRA decreased smoothly though the algorithm converged quite slow. So we can infer that HSIRA did not take bad restarting vectors. Remarkably, the figure and table tell us that RHSIRA converged much faster. This is due to the fact that the refined harmonic Ritz vector $\hat{y}$ can be much more accurate than the corresponding harmonic Ritz vector $y$, so that restarting vectors were better and the subspaces generated were more accurate as restarts proceeded.

**Example 3.** This unsymmetric eigenproblem dw4096 of $n = 8192$ arises from dielectric channel waveguide problems [1]. We test this example with $\sigma = -24$ and Max $= 500$. The computed eigenvalue is $\lambda \approx -30.217$. The preconditioner $M$ is the incomplete LU factorization of $A - \sigma I$ with drop tolerance $10^{-3}$. Figure 3 and Table 3 display the results.

| Accuracy | Algorithm | $I_{restart}$ | $I_{inner}$ | $P_{0;1}$ | Algorithm | $I_{restart}$ | $I_{inner}$ | $P_{0;1}$ |
|----------|-----------|---------------|-------------|-----------|-----------|---------------|-------------|-----------|
| $\tilde{\varepsilon} = 10^{-3}$ | SIRA | Max | 15722 | 45% | JD | Max | 17118 | 45% |
| | HSIRA | 306 | 9473 | 85% | HJD | 300 | 9246 | 85% |
| | RHSIRA | 168 | 5019 | 96% | RHJD | 151 | 4540 | 96% |
| $\tilde{\varepsilon} = 10^{-4}$ | SIRA | Max | 20686 | 14% | JD | Max | 20794 | 14% |
| | HSIRA | 289 | 9251 | 49% | HJD | 282 | 9091 | 49% |
| | RHSIRA | 156 | 4722 | 94% | RHJD | 197 | 5952 | 95% |
| exact | SIRA | Max | 87410 | – | JD | Max | 89435 | – |
| | HSIRA | 255 | 44906 | – | HJD | 259 | 45480 | – |
| | RHSIRA | 127 | 23351 | – | RHJD | 139 | 25419 | – |

Table 3: Example 3. dw4096 with $\sigma = -24$.

Figure 3 indicates that the exact and inexact SIRA oscillated sharply in the first cycle, the exact and inexact HSIRA improved the situation very significantly but still did not behave quite regularly, while the exact and inexact RHSIRA converged very smoothly after first a few outer iterations and considerably faster than the HSIRA. Poor Ritz vectors further led to poor subspace expansion vectors, generating a sequence of poor subspaces. We observed very similar phenomena in the JD type algorithms. This means that the standard Ritz vectors were not suitable for restarting, and the harmonic Ritz vectors were much better but inferior.
Figure 3: Example 3. dw4096 with $\sigma = -24$.

to the refined harmonic Ritz vectors for restarting. So it is expected that the restarted exact and inexact SIRA and JD algorithms may not work well, but the restarted HSIRA and HJD may work much better than the former ones and the restarted RHSIRA and RHJD outperform the HSIRA and HJD very considerably.

The above expectations are confirmed by and Table 3 and the bottom-left subfigure of Figure 3. It is seen that the restarted exact and inexact SIRA and JD algorithms failed to converge within 500 restarts while HSIRA and HJD solved the problem successfully and the refined RHSIRA and RHJD were twice as efficient as the harmonic algorithms, as indicated by $I_{\text{restart}}$’s. Furthermore, we see the inexact algorithms with $\tilde{\varepsilon} = 10^{-3}, 10^{-4}$ exhibited very similar convergence behavior and mimic the exact algorithms very well. This confirms our theory that modest $\tilde{\varepsilon}$ is generally enough to make the inexact SIRA and JD type algorithms behave very like the corresponding exact counterparts and smaller $\tilde{\varepsilon}$ is not necessary. We also find that each exact SIRA type algorithm is as efficient as the corresponding JD type algorithm for the same $\tilde{\varepsilon}$.

Regarding the overall performance, the comments on Examples 1–2 apply here analogously. For $\tilde{\varepsilon} = 10^{-3}$ and $10^{-4}$, the inexact HSIRA, RHSIRA, HJD and RHJD were four times faster than the corresponding exact algorithms. Furthermore, the restarted exact and inexact RHSIRA and RHJD were twice as fast as the corresponding HSIRA and HJD algorithms for the same $\tilde{\varepsilon}$.

Example 4. This unsymmetric eigenproblem dw8192 of $n = 8192$ arises from dielectric channel waveguide problems [1]. We test this example with $\sigma = -60$ and $\text{Max} = 1000$. The computed eigenvalue is $\lambda \approx -87.795$. The preconditioner $M$ is the incomplete LU factorization of $A - \sigma I$ with drop tolerance $10^{-3}$. Figure 4 and Tables 4 report the results.
Figure 4: Example 4. $dw8192$ with $\sigma = -60$.

| $\tilde{\varepsilon}$ | Algorithm | $I_{\text{restart}}$ | $I_{\text{inner}}$ | $P_{0.1}$ | Algorithm | $I_{\text{restart}}$ | $I_{\text{inner}}$ | $P_{0.1}$ |
|------------------------|-----------|----------------------|--------------------|-----------|-----------|----------------------|--------------------|-----------|
| $10^{-3}$              | SIRA      | Max 29262            | 39%                | JD        | Max 32222 | 39%                |
|                        | HSIRA     | 542                  | 15953              | 76%       | HJD       | 402                 | 11848              | 81%       |
|                        | RHSIRA    | 372                  | 10813              | 96%       | RHJD      | 317                 | 9221               | 96%       |
| $10^{-4}$              | SIRA      | Max 38502            | 12%                | JD        | Max 39413 | 12%                |
|                        | HSIRA     | 555                  | 17416              | 51%       | HJD       | 497                 | 15691              | 57%       |
|                        | RHSIRA    | 386                  | 11595              | 93%       | RHJD      | 429                 | 12884              | 93%       |
| exact                  | SIRA      | Max 145435           | –                  | JD        | Max 151453 | –                  |
|                        | HSIRA     | 615                  | 89636              | –         | HJD       | 617                 | 90056              | –         |
|                        | RHSIRA    | 387                  | 59170              | –         | RHJD      | 383                 | 59166              | –         |

Table 4: Example 4. $dw8192$ with $\sigma = -60$.

All the algorithms gave results that are typically similar to those corresponding counterparts obtained for Example 3. The comments and conclusions are very analogous, and we thus omit them.

6 Conclusions

The standard SIRA and JD methods may not work well for the interior eigenvalue problem. The standard Ritz vectors may converge irregularly, and it is also hard to select correct Ritz pairs to approximate the desired interior eigenpairs. So the Ritz vectors may not be good choices for restarting, causing that restarted algorithms may perform poorly. Meanwhile, the Ritz vectors appear to expand subspaces poorly. In contrast, the harmonic Ritz vectors
are more regular and more reliable approximations to the desired eigenvectors, so that they may expand the subspaces better and generate more accurate a sequence of subspaces when restarting the methods. Due to the optimality, the refined harmonic Ritz vectors are generally more accurate than the harmonic ones and are better choices for expanding subspaces or restarting the methods. Most importantly, we have proved that the harmonic and refined SIRA and JD methods generally mimic their exact counterparts well provided that all inner linear systems are solved with only low or modest accuracy. To be practical, we have presented the restarted harmonic and refined harmonic SIRA and JD algorithms. Meanwhile, we have designed practical stopping criteria for inner iterations.

Numerical experiments have confirmed our theory. They have indicated that the restarted harmonic and refined harmonic SIRA and JD algorithms are much more efficient than the restarted standard SIRA and JD algorithms for the interior eigenvalue problem. Furthermore, the refined harmonic algorithms are much better than the harmonic ones. Importantly, the results have demonstrated that each inexact algorithm behaves like its exact counterpart when all inner linear systems are solved with only low or modest accuracy. In addition, numerical results have confirmed that each exact or inexact SIRA type algorithm is equally as efficient as the corresponding JD type algorithm for the same $\tilde{\varepsilon}$.

Our algorithms are designed to compute only one interior eigenvalue and its associated eigenvector. This is a special nature of SIRA and JD type methods: they compute only one eigenpair each time. If more than one eigenpairs are of interest, we may extend the algorithms to this situation in a number of ways. For example, we may introduce some deflation techniques into them. Also, we may change the target $\sigma$ to a new one, to which the second desired eigenvalue is the closest, and apply the algorithms to find it and its associated eigenvector. Proceed this way until all the desired eigenpairs are found. Such kind of algorithms is under developments.

References

[1] B. Boisvert, R. Pozo, K. Remington, B. Miller, and R. Lipman, Matrix Market, available online at http://math.nist.gov/MatrixMarket/, 2004.
[2] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide, SIAM, Philadelphia, PA, 2000.
[3] M. Crouzeix, B. Philippe and M. Sadkane, The Davidson method, SIAM J. Sci. Comput., 15 (1994), pp. 62–70.
[4] E. R. Davidson, The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices, J. Comput. Phys., 17 (1975), pp. 87–94.
[5] M. A. Freitag and A. Spence, Shift-and-invert Arnoldi’s method with preconditioned iterative solvers, SIAM J. Matrix Anal. Appl., 31 (2009), pp. 942–969.
[6] M. E. Hochstenbach and Y. Notay, Controlling inner iterations in the Jacobi–Davidson method, SIAM J. Matrix Anal. Appl., 31 (2009), pp. 460–477.
[7] Z. Jia, The convergence of generalized Lanczos methods for large unsymmetric eigenproblems, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 843–862.
[8] Z. Jia, Refined iterative algorithms based on Arnoldi’s process for large unsymmetric eigenproblems, Linear Algebra Appl., 259 (1997), pp. 1–23.
[9] Z. Jia, A refined subspace iteration algorithm for large sparse eigenproblems, Appl. Numer. Math, 32 (2000), pp. 35–52.
[10] Z. Jia, The refined harmonic Arnoldi method and an implicitly restarted refined algorithm for computing interior eigenpairs, Appl. Numer. Math, 42 (2002), pp. 489–512.
[11] Z. Jia, Some theoretical comparisons of refined Ritz vectors and Ritz vectors, Sci. China Ser. A, 47 (Suppl.), (2004), pp. 222–233.

[12] Z. Jia, The convergence of harmonic Ritz values, harmonic Ritz vectors, and refined harmonic Ritz vectors, Math. Comput., 74 (2005), pp. 1441–1456.

[13] Z. Jia, Using cross-product matrices to compute the SVD, Numer. Algor., 42 (2006), pp. 31–61.

[14] Z. Jia and C. Li, On inner iterations in the shift-invert residual Arnoldi method and the Jacobi–Davidson method, arXiv:1109.5455v3, 2012.

[15] Z. Jia and G. W. Stewart, An analysis of the Rayleigh-Ritz method for approximating eigenspaces, Math. Comput., 70 (2001), pp. 637–648.

[16] C. Lee and G. W. Stewart, Analysis of the residual Arnoldi method, TR-4890, Department of Computer Science, University of Maryland at College Park, 2007.

[17] R. B. Morgan, Computing interior eigenvalues of large matrices, Linear Algebra Appl., 154-156 (1991), pp. 289–309.

[18] R. B. Morgan and M. Zeng, Harmonic projection methods for large non-symmetric eigenvalue problems, Numer. Linear Algebra Appl., 5 (1998), pp. 33–55.

[19] B. N. Parlett, The Symmetric Eigenvalue Problem, SIAM, Philadelphia, PA, 1998.

[20] Y. Saad, Numerical Methods for Large Eigenvalue Problems, Manchester University Press, UK, 1992.

[21] V. Simoncini, Variable accuracy of matrix-vector products in projection methods for eigencomputation, SIAM J. Numer. Anal., 43 (2005), pp. 1155–1174.

[22] 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.

[23] G. Sleijpen and H. Van der Vorst, A Jacobi-Davidson iteration method for linear eigenvalue problems, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 401–425. Reprinted in SIAM Review, (2000), pp. 267–293.

[24] A. Statthopoulous and Y. Saad, Restarting techniques for the (Jacobi-)Davidson symmetric eigenvalue methods, Electron Trans. Numer. Anal., 7 (1998), pp. 163–181.

[25] G. W. Stewart, Matrix Algorithms, Vol II: Eigensystems, SIAM, Philadelphia, PA, 2001.

[26] H. Van der Vorst, Computational Methods for Large Eigenvalue Problems, Elsevier, North Holland, 2002.