SIAM J. SCI. COMPUT.
Vol. 41, No. 3, pp. A1574–A1603

ON INNER ITERATIONS OF JACOBI–DAVIDSON TYPE METHODS FOR LARGE SVD COMPUTATIONS[*]

JINZHI HUANG[†] AND ZHONGXIAO JIA[†]

Abstract. We make a convergence analysis of the harmonic and refined harmonic extraction versions of Jacobi–Davidson SVD (JDSVD) type methods for computing one or more interior singular triplets of a large matrix A. At each outer iteration of these methods, a correction equation, i.e., inner linear system, is solved approximately by using iterative methods, which leads to two inexact JDSVD type methods, as opposed to the exact methods where correction equations are solved exactly. Accuracy of inner iterations critically affects the convergence and overall efficiency of the inexact JDSVD methods. A central problem is how accurately the correction equations should be solved to ensure that both of the inexact JDSVD methods can mimic their exact counterparts well, that is, they use almost the same outer iterations to achieve the convergence. In this paper, similar to the available results on the JD type methods for large matrix eigenvalue problems, we prove that each inexact JDSVD method behaves like its exact counterpart if all the correction equations are solved with *low* or *modest* accuracy during the outer iterations. Based on the theory, we propose practical stopping criteria for inner iterations. Numerical experiments confirm our theory and the effectiveness of the inexact algorithms.

Key words. singular triplets, JDSVD method, harmonic extraction, refined harmonic extraction, subspace expansion, inner iteration

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

DOI. 10.1137/18M1192019

1. Introduction.

Consider the singular value decomposition (SVD)

$$\begin{cases} Av_i = \sigma_i u_i, \\ A^T u_i = \sigma_i v_i, \end{cases} \qquad i = 1, 2, \ldots, N, \tag{1.1}$$

of a large and possibly sparse matrix $A \in \mathbb{R}^{M\times N}$ with $M \geq N$. The $(\sigma_i, u_i, v_i),\ i = 1, 2, \ldots, N$, are called the singular triplets of A. For a given target $\tau > 0$, let the singular values $\sigma_1, \sigma_2, \ldots, \sigma_N$ be labeled as

$$|\sigma_1 - \tau| < |\sigma_2 - \tau| \leq \cdots \leq |\sigma_N - \tau|. \tag{1.2}$$

We are interested in the simple singular value σ_1 closest to the target τ and/or the corresponding left and right singular vectors u_1 and v_1. We will denote (σ_1, u_1, v_1) by (σ, u^*, v^*) for simplicity, which is called an interior singular triplet when τ is inside the singular spectrum of A. We make two remarks. The first is that for all the algorithms to be proposed, the analysis and results apply to a complex A trivially with the transpose of a vector or matrix replaced by its conjugate transpose. The second is that we focus on the analysis of the algorithms for computing one simple singular triplet of A and then derive the algorithms with deflation for computing more than one, i.e., $\ell > 1$, singular triplet of A and extend our analysis and results to these variants.

[*] Submitted to the journal's Methods and Algorithms for Scientific Computing section June 4, 2018; accepted for publication (in revised form) March 19, 2019; published electronically May 16, 2019.
http://www.siam.org/journals/sisc/41-3/M119201.html

Funding: This work was supported in part by the National Science Foundation of China through grant 11771249.

[†] Department of Mathematical Sciences, Tsinghua University, 100084 Beijing, China (huangjz15@mails.tsinghua.edu.cn, jiazx@tsinghua.edu.cn).

Projection methods that are widely used for large eigenproblems [2, 20, 22, 25] have been adapted to the computation of (σ, u^*, v^*). Given projection subspaces, there are four extraction approaches: standard, harmonic, refined, and refined harmonic extractions. The standard extraction [4, 5, 6, 14, 23, 26] is well suited for large singular values (cf. [5, Theorem 4.3]). The harmonic extraction is preferable to the interior and the smallest singular values. However, the approximate singular vectors obtained by the standard and harmonic extractions may converge irregularly or even fail to converge even though the approximate singular values converge [13, 14]. The refined and refined harmonic extractions [5, 6, 8, 10, 14, 15, 17, 18, 27] fix the possible nonconvergence of singular vectors and obtain more accurate approximations. As observed and claimed in [6, 14, 15, 17, 28, 29], the refined and refined harmonic extractions appear to give better accuracy than the standard and harmonic counterparts, respectively.

As a matter of fact, when computing an interior singular triplet (σ, u^*, v^*), that is, when τ lies inside the spectrum of singular values of A, the standard extraction encounters a serious difficulty even if searching subspaces are sufficiently good: it is practically hard to pick up a correct Ritz value to approximate σ^* even though there is a good one among them. As a result, whenever a wrong Ritz value is selected, the refined extraction definitively fails to deliver a good approximation to (u^*, v^*). Therefore, the refined extraction may also be unsuitable to compute the truly interior (σ, u^*, v^*). In contrast, the above two difficulties can be nicely overcome by using the harmonic and refined harmonic extractions, especially by the latter one because the refined harmonic extraction will produce better approximate singular vectors whose convergence can be guaranteed provided that searching subspaces are sufficiently accurate [13, 27]. In other words, unlike the standard and refined extractions, for good searching subspaces, the harmonic and refined harmonic extractions produce good approximations to the interior (σ^*, u^*, v^*), and the refined harmonic extraction is favorable due to its better convergence. Because of these reasons, we will only consider the harmonic and refined harmonic extractions in this paper.

Sleijpen and van der Vorst [24] propose the Jacobi–Davidson (JD) method for large matrix eigenvalue problems, which uses the standard and harmonic extractions. Hochstenbach [5, 6] extends the JD method to the SVD computation in a novel way, referred to as JDSVD, which requires that an approximate singular triplet satisfies a certain double orthogonality condition rather than the ordinary orthogonality condition. Moreover, unlike the JD method applied to the eigenvalue problem of the augmented matrix $\left[\begin{smallmatrix} 0 & A \\ A^T & 0 \end{smallmatrix}\right]$ directly, JDSVD enlarges its projection subspace by two dimensions rather than one dimension as done in JD. Obviously, JDSVD is mathematically different from the JD method applied to the augmented matrix directly and is shown to be more suitable and more effective than the direct application of the JD method for the eigenvalue problem to the mentioned augmented matrix. Wu and Stathopoulos [29] propose a two-stage SVD method, called PHSVD, which transforms the SVD computation into the eigenvalue problems of A^TA and $\left[\begin{smallmatrix} 0 & A \\ A^T & 0 \end{smallmatrix}\right]$. Unlike JDSVD, PHSVD performs the projection first on the eigenvalue problem of the cross-product A^TA and then uses the JD type methods to solve the eigenvalue problem of $\left[\begin{smallmatrix} 0 & A \\ A^T & 0 \end{smallmatrix}\right]$. At each iteration of JDSVD or the second stage of PHSVD, one needs to solve a correction equation when expanding searching subspaces. Wu, Romero, and Stathopolous [28] develop the software PRIMME_SVDS that implements the PHSVD method using the standard extraction and the refined or refined harmonic extraction, and they provide intuitive user interfaces in C, MATLAB, Python, and R for both ordinary and advanced users to fully exploit the power of PRIMME_SVDS.

Since the matrix A is large, for each method it is generally impractical to solve the correction equation by direct solvers. Therefore, the correction equation has to be solved approximately by iterative solvers. This gives rise to the inexact JD type methods for either eigenvalue problems or SVD computations, which distinguish them from the exact methods where all the correction equations are solved exactly. Accuracy of inner iterations plays the critical role in the convergence and overall efficiency of the inexact JD methods. Naturally, a central question is how accurately the correction equations should be solved in order to ensure that they use almost the same outer iterations to achieve the desired convergence.

Accuracy issues on inner iterations in the JD type methods are fundamental and have been receiving intensive attention for the JD type methods for large matrix eigenvalue problems since the advent of the JD method. Notay [19] makes an analysis on the *simplified* JD method, i.e., without subspace acceleration, for the eigenvalue problem, called the JDCG algorithm, in which the correction equation is solved by the preconditioned CG method and the correction vector is directly added to the current approximate eigenvector to form a new approximation in the next outer iteration. He shows that during inner iterations, the convergence of the outer iteration toward the desired eigenpair can be monitored by the reduction of the residual norms of inner iterations. Based on this result, Notay proposes stopping criteria for inner iterations of the JDCG algorithm. Hochstenbach and Notay [7] extend the results in [19] to the generalized Hermitian eigenvalue problem. Wu, Romero, and Stathopolous [28, 29] adapt the stopping criteria in [19] for inner iterations to their general JD type methods for large SVD computations, in which they use the quasi-minimal residual (QMR) algorithm rather than the CG algorithm to solve generally indefinite correction equations. The JDSVD methods in [5, 6] simply stop inner iterations after a small number of iterations are performed.

Unfortunately, Notay's analysis approach and the results in [7, 19, 28, 29] are only applicable to the simplified JD type methods, and they cannot be extended to the more complicated and practical *general* JD type methods with subspace acceleration for the eigenvalue problem or the SVD computation, in which the (approximate) solution of the correction equation is used to expand the subspace and a new approximate eigenpair or singular triplet is formed by performing one of the four extractions onto the expanded subspace. However, one must be aware that the simplified JD method is seldom used practically due to its low effectiveness. For the general JD type methods for eigenvalue problems that are based on standard, harmonic, and refined harmonic extractions, Jia and Li [12, 13] prove for the first time that once all the correction equations are solved only with *low or modest* accuracy by any iterative solver, the JD type methods can mimic their exact counterparts well, in the sense that the outer iterations used by the former ones to achieve the convergence are almost the same as those used by the latter ones. Based on the theory established, they propose practical stopping criteria for inner solvers. A remarkable fact is that, whenever the JD type methods for the SVD computation are mathematically equivalent to the JD type methods applied to the eigenvalue problem of $\left[\begin{smallmatrix} 0 & A \\ A^T & 0 \end{smallmatrix}\right]$, all the results in [12, 13] directly carry over to them, which include the *relevant* algorithms in [28, 29].

However, the results in [12, 13] have not yet been extended to the double orthogonality based JDSVD methods in [5, 6]. It is unclear whether or not similar results exist and are applicable to these JDSVD methods. Therefore, reliable and general-purpose stopping criteria for corresponding inner iterations are still lacking.

In this paper, by requiring that an approximate singular triplet satisfies the double orthogonality condition [5], we propose a harmonic extraction based JDSVD type method, in which the approximate singular value must be a certain Rayleigh quotient rather than the associated harmonic Ritz value. Then, by combining the harmonic extraction with the refined extraction, we present a refined harmonic extraction based JDSVD type method. Our harmonic and refined harmonic extractions are mathematically different from the ones in [28, 29] in that (i) ours require a double orthogonality condition while theirs use the ordinary orthogonality and (ii) ours expand the projection subspace by two dimensions while theirs expand the projection subspace by one dimension at each outer iteration. Our methods and their derivations are also different from the harmonic and refined extraction based JDSVD methods in [5, 6] in several respects; for example, the latter ones use the principle of harmonic and refined extractions in different manners. On the other hand, we *observe* a key fact, specifically that the correction equations involved in our methods and the ones in [5, 6] at each outer iteration *have the same form*. Therefore, the *topic* of solution accuracy of the relevant correction equations can be considered in a unified manner.

Precisely, our primary goal is to establish the *lowest* accuracy requirement for inner iterations involved in JDSVD methods such that each inexact method mimics its exact counterpart well, that is, they use a comparable number of outer iterations to achieve the desired level of accuracy. The goal is fundamental. First, with the exact JDSVD type methods as a reference standard, it guarantees the robustness and generality of the corresponding inexact counterparts. Second, under this presupposition, the inexact JDSVD type algorithms are the most efficient since, for a given iterative solver, we can use the least cost, i.e., fewest inner iterations, to solve each correction equation at every outer iteration.

Inspired by the work of Jia and Li [12, 13], under the assumption that the current approximate singular triplet is reasonably good, we make a rigorous one-step analysis on the two inexact JDSVD type algorithms. We first establish an intimate connection between the solution accuracy of the correction equation and the accuracy of the expansion vectors. Then we derive the accuracy relationship between the inexact and exact expanded subspaces, based on which we establish the accuracy requirement for inner iterations. The results show that each inexact JDSVD mimics the corresponding exact JDSVD well once all the correction equations are solved with modest or low accuracy, that is, ranging between 10^{-4} and 10^{-3}, during the outer iterations. These results are similar to those on the inexact JD type methods for general eigenvalue problems [12, 13]. Furthermore, we prove that these results can be extended to the JDSVD type methods with deflation that are used to compute $\ell > 1$ singular triplets of A. We consider practical issues and design practical stopping criteria for inner iterations in the JDSVD methods. It is important to recall that all the results hold for the proposed methods in this paper and the methods in [5, 6] because the correction equations in them have the same form, as mentioned previously. Finally, we report numerical experiments to justify that our theory works well.

The paper is organized as follows. In section 2, we describe the harmonic and refined harmonic JDSVD methods. In section 3, we derive relationships between the solution accuracy of the correction equation and the accuracy of the expansion vectors. We prove that the inexact JDSVD methods can mimic the exact JDSVD methods well when the solution accuracy of the correction equation is low or modest, i.e., $10^{-4} \sim 10^{-3}$, during all outer iterations. In section 4, we describe the JDSVD type methods with deflation for computing more than one singular triplet and extend our results to this case. In section 5, we design practical stopping criteria for inner iterative solvers. In section 6, we report numerical experiments to confirm our theory. Finally, we conclude the paper in section 7.

Throughout the paper, denote by $\|\cdot\|$ the 2-norm and $\kappa(X)=\sigma_{\max}(X)/\sigma_{\min}(X)$ the condition number of a matrix X with $\sigma_{\max}(X)$ and $\sigma_{\min}(X)$ being the largest and smallest singular values of X, by I_k the identity matrix of order k, and by X^T the transpose of X.

2. Harmonic and refined harmonic JDSVD methods.

Assume that we have a pair of m-dimensional searching subspaces $\mathcal{U}$ and $\mathcal{V}$, from which we compute an approximate singular triplet (θ, u, v) satisfying $u \in \mathcal{U}, v \in \mathcal{V}$ and $\|u\| = \|v\| = 1$. Define the residual of (θ, u, v) as

$$r = r(\theta, u, v) := \begin{bmatrix} Av - \theta u \\ A^T u - \theta v \end{bmatrix}. \tag{2.1}$$

We require that $r \perp\perp (u, v)$, where $\perp\perp$ means that $Av - \theta u \perp u$ and $A^T u - \theta v \perp v$, which is called double orthogonality in [5]. We mention that the so-called ordinary orthogonality condition means $r \perp [u^T, v^T]^T$.

Assume that the columns of U and V form orthonormal bases of $\mathcal{U}$ and $\mathcal{V}$, respectively, and (θ, u, v) is reasonably good but not converged yet. Then a JDSVD type method expands $\mathcal{U}$ and $\mathcal{V}$ in the following way:

Solve the correction equation

$$\begin{bmatrix} I_M - uu^T & \\ & I_N - vv^T \end{bmatrix} \begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix} \begin{bmatrix} I_M - uu^T & \\ & I_N - vv^T \end{bmatrix} \begin{bmatrix} s \\ t \end{bmatrix} = -r \tag{2.2}$$

for $(s, t) \perp\perp (u, v)$, where $s \in \mathbb{R}^M$ and $t \in \mathbb{R}^N$. Orthonormalize s and t against U and V to get u_+ and v_+, respectively. The normalized subspace expansion vectors are

$$u_+ = \frac{(I_M - P_U)s}{\|(I_M - P_U)s\|}, \qquad v_+ = \frac{(I_N - P_V)t}{\|(I_N - P_V)t\|}, \tag{2.3}$$

where $P_U = UU^T$ and $P_V = VV^T$ are the orthogonal projectors onto $\mathcal{U}$ and $\mathcal{V}$, respectively. The columns of $U_+ = [U, u_+]$ and $V_+ = [V, v_+]$ are the orthonormal bases of the expanded subspaces $\mathcal{U}_+$ and $\mathcal{V}_+$, from which we can compute new approximate singular triplet.

2.1. Harmonic Jacobi–Davidson SVD method.

It is known that $(\sigma, w^* = \frac{1}{\sqrt{2}} \left[\begin{smallmatrix} u^* \\ v^* \end{smallmatrix}\right])$ is an eigenpair of the augmented matrix

$$C = \begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix}. \tag{2.4}$$

We use the harmonic extraction described in [13, 18] to compute an approximation of (σ, w^*). Notice that the projection or searching subspace is spanned by the columns of $\left[\begin{smallmatrix} U & \\ & V \end{smallmatrix}\right]$. Given the target $\tau > 0$, let the approximate eigenvector be $\widetilde{w} = \left[\begin{smallmatrix} U & \\ & V \end{smallmatrix}\right] \widetilde{f} \in \mathbb{R}^{M+N}$, and define

$$G = \begin{bmatrix} U^T & \\ & V^T \end{bmatrix} \begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix}^2 \begin{bmatrix} U & \\ & V \end{bmatrix} \tag{2.5}$$

and

$$F = \begin{bmatrix} U^T & \\ & V^T \end{bmatrix} \begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix} \begin{bmatrix} U & \\ & V \end{bmatrix} = \begin{bmatrix} -\tau I_m & H \\ H^T & -\tau I_m \end{bmatrix} \tag{2.6}$$

with $H = U^T AV$. Then the harmonic extraction on C amounts to solving the following generalized symmetric eigenvalue problem of the matrix pencil (F, G):

$$\begin{cases} F\widetilde{f} = \dfrac{1}{\nu} G\widetilde{f} \text{ with } \|\widetilde{f}\| = 1, \\ \vartheta = \nu + \tau \end{cases} \tag{2.7}$$

for the largest eigenvalue $\frac{1}{\nu}$ in magnitude and the corresponding eigenvector $\widetilde{f} = \left[\widetilde{c}^T, \widetilde{d}^T\right]^T$, where $\widetilde{c}, \widetilde{d} \in \mathbb{R}^m$. We take $(|\vartheta|, \widetilde{u} = U\frac{\widetilde{c}}{\|\widetilde{c}\|},\ \widetilde{v} = V\frac{\widetilde{d}}{\|\widetilde{d}\|})$ to approximate (σ, u^*, v^*) if $\widetilde{u}^T A\widetilde{v} \geq 0$; otherwise, we take $(|\vartheta|, \widetilde{u} = U\frac{\widetilde{c}}{\|\widetilde{c}\|},\ \widetilde{v} = -V\frac{\widetilde{d}}{\|\widetilde{d}\|})$.

If $(|\vartheta|, \widetilde{u}, \widetilde{v})$ has not yet converged, we need to expand the subspaces $\mathcal{U}$ and $\mathcal{V}$. Let $u = \widetilde{u}$ and $v = \widetilde{v}$, and replace $|\vartheta|$ by the Rayleigh quotient

$$\theta = \rho = |\widetilde{u}^T A\widetilde{v}| = \frac{|\widetilde{c}^T U^T AV\widetilde{d}|}{\|\widetilde{c}\|\|\widetilde{d}\|} = \frac{|\widetilde{c}^T H\widetilde{d}|}{\|\widetilde{c}\|\|\widetilde{d}\|}. \tag{2.8}$$

Then we obtain a consistent equation (2.2) with $r = r(\rho, \widetilde{u}, \widetilde{v})$. We use the solution vectors s and t to compute the expansion vectors u_+ and v_+ defined by (2.3) and obtain $\mathcal{U}_+$ and $\mathcal{V}_+$. The new projection subspace is spanned by the columns of $\left[\begin{smallmatrix} U_+ & \\ & V_+ \end{smallmatrix}\right]$, and its dimension is thus increased by two. The resulting method is called the harmonic Jacobi–Davidson SVD (HJDSVD) method.

We mention in passing that in the JD type methods applied to the eigenvalue problem of $\left[\begin{smallmatrix} 0 & A \\ A^T & 0 \end{smallmatrix}\right]$ directly, the current projection subspace is spanned by the orthonormal columns of $[U^T, V^T]^T$ and the expanded one is spanned by those of $[U_+^T, V_+^T]^T$, whose dimension is increased by one. Another mathematically essential distinction is that the whole expanding vector $[u_+^T, v_+^T]^T$ is obtained by orthonormalizing $[s^T, t^T]^T$ against the column orthonormal $[U^T, V^T]^T$, where $[s^T, t^T]^T$ is the solution to a corresponding correction equation in a *different* form from (2.2).

It is easily justified by the optimality of the Rayleigh quotient that ρ is more accurate than $|\vartheta|$ in the sense that $\|r(\rho, \widetilde{u}, \widetilde{v})\| \leq \|r(|\vartheta|, \widetilde{u}, \widetilde{v})\|$. Notice that in computation, by (2.8) we will take $\widetilde{u} = U\frac{\widetilde{c}}{\|\widetilde{c}\|},\ \widetilde{v} = V\frac{\widetilde{d}}{\|\widetilde{d}\|}$ for $\widetilde{c}^T H\widetilde{d} \geq 0$ and $\widetilde{u} = U\frac{\widetilde{c}}{\|\widetilde{c}\|},\ \widetilde{v} = -V\frac{\widetilde{d}}{\|\widetilde{d}\|}$ for $\widetilde{c}^T H\widetilde{d} < 0$. This guarantees that $\rho = \widetilde{u}^T A\widetilde{v} \geq 0$ and that $\widetilde{u}$ and $\widetilde{v}$ are correct approximate left and right singular vectors.

2.2. Refined harmonic Jacobi–Davidson SVD method. Assume that we have already obtained an approximate singular value ρ of A using the HJDSVD method. With ρ available, we can use the refined extraction to compute new and better approximations to u^* and v^*. Such an approach is called the refined harmonic extraction, which seeks a unit vector $\widehat{w} = \left[\begin{smallmatrix} U & \\ & V \end{smallmatrix}\right] \widehat{f} \in \mathbb{R}^{M+N}$ such that

$$\|C\widehat{w} - \rho\widehat{w}\| = \min_{w \in span\{U,V\}, \|w\|=1} \|Cw - \rho w\|, \tag{2.9}$$

where C is defined as (2.4) and span$\{U, V\}$ denotes the range of $\left[\begin{smallmatrix} U & \\ & V \end{smallmatrix}\right]$.

For the approximate eigenvector $\widetilde{w}$ obtained by HJDSVD, it has been proved by Jia [9] that $\|(C - \rho I)\widehat{w}\| < \|(C - \rho I)\widetilde{w}\|$, provided that $(\rho, \widetilde{w})$ is not an exact eigenpair of C; moreover, if $(\rho, \widetilde{w})$ is not an exact eigenpair of C and there is another harmonic Ritz value of C close to ρ, then it may occur that

$$\|(C - \rho I)\widehat{w}\| \ll \|(C - \rho I)\widetilde{w}\|,$$

meaning that $\widehat{w}$ can be a much more accurate approximation to $w^* = \frac{1}{\sqrt{2}} \left[\begin{smallmatrix} u^* \\ v^* \end{smallmatrix}\right]$.

Equation (2.9) is equivalent to

$$\left\|\begin{bmatrix}-\rho U & AV\\ A^TU & -\rho V\end{bmatrix}\widehat{f}\right\| = \min_{f\in\mathbb{R}^{2m},\|f\|=1}\left\|\begin{bmatrix}-\rho U & AV\\ A^TU & -\rho V\end{bmatrix}f\right\|. \tag{2.10}$$

Therefore, $\widehat{f}$ is the right singular vector of $D = \left[\begin{smallmatrix}-\rho U & AV\\ A^TU & -\rho V\end{smallmatrix}\right]$ corresponding to its smallest singular value, that is, $\widehat{f}$ is an eigenvector of

$$G' = D^TD = \begin{bmatrix}U^TAA^TU+\rho^2I_m & -2\rho H\\ -2\rho H^T & V^TA^TAV+\rho^2I_m\end{bmatrix} \tag{2.11}$$

associated with its smallest eigenvalue. Jia [11] proves that the smallest singular value and associated right singular vector $\widehat{f}$ of D can be computed accurately to the working precision by applying the QR algorithm to the eigenvalue problem of the cross-product matrix G', provided that the smallest singular value of D is not close to its second smallest one. So instead of computing the SVD of D, we work on G' and compute its eigenvector $\widehat{f}$. In this way, we reduce the cost substantially by solving a small $2m\times 2m$ matrix eigenvalue problem of G' rather than computing the SVD of the $(M+N)\times 2m$ matrix D.

Let $\widehat{f} = \left[\widehat{c}^T, \widehat{d}^T\right]^T$, where $\widehat{c},\ \widehat{d}\in\mathbb{R}^m$. We take $\widehat{u} = U\frac{\widehat{c}}{\|\widehat{c}\|}$ and $\widehat{v} = V\frac{\widehat{d}}{\|\widehat{d}\|}$ (or $\widehat{v} = -V\frac{\widehat{d}}{\|\widehat{d}\|}$) as new approximations to u^* and v^*, respectively.

If $(\rho,\widehat{u},\widehat{v})$ has not yet converged, we need to expand $\mathcal{U}$ and $\mathcal{V}$ to $\mathcal{U}_+$ and $\mathcal{V}_+$ in the same manner as done in HJDSVD. Since the consistency of the correction equation (2.2), where $u=\widehat{u}$ and $v=\widehat{v}$, requires the double orthogonality of the residual r, we take the Rayleigh quotient

$$\theta = \rho' = |\widehat{u}^TA\widehat{v}| = \frac{|\widehat{c}^TU^TAV\widehat{d}|}{\|\widehat{c}\|\|\widehat{d}\|} = \frac{|\widehat{c}^TH\widehat{d}|}{\|\widehat{c}\|\|\widehat{d}\|} \tag{2.12}$$

and compute the residual $r = r(\rho',\widehat{u},\widehat{v})$. The solution vectors s and t are then used to expand $\mathcal{U}$ and $\mathcal{V}$, respectively, and the resulting method is called the refined harmonic Jacobi–Davidson SVD (RHJDSVD) method.

Similar to HJDSVD, we take $\widehat{u} = U\frac{\widehat{c}}{\|\widehat{c}\|}$ and $\widehat{v} = V\frac{\widehat{d}}{\|\widehat{d}\|}$ for $\widehat{c}^TH\widehat{d}\geq 0$ and $\widehat{u} = U\frac{\widehat{c}}{\|\widehat{c}\|}$, $\widehat{v} = -V\frac{\widehat{d}}{\|\widehat{d}\|}$ for $\widehat{c}^TH\widehat{d} < 0$, which guarantees that $\rho' = \widehat{u}^TA\widehat{v}\geq 0$ and that $\widehat{u}$ and $\widehat{v}$ are correct approximate left and right singular vectors.

Algorithm 1 describes the HJDSVD and RHJDSVD methods, where, and in the next section, to unify our presentation, we denote by (θ,u,v) the current approximation $(\rho,\widetilde{u},\widetilde{v})$ or $(\rho',\widehat{u},\widehat{v})$ to (σ,u^*,v^*).

3. A convergence analysis.

In this section, we make a convergence analysis on the inexact HJDSVD and RHJDSVD methods. We should stress that in the analysis of this section, we have no restriction to the way that an approximate singular triplet (θ,u,v) is extracted, which means that all our analysis of this section is applicable to the standard and refined extraction based JDSVD methods. Before proceeding, we must point out that our analysis is asymptotic and a rigorous one-step one, i.e., it is a *local* rather than a *global* analysis since we must assume that the current (θ,u,v) is reasonably good but not yet converged.

Algorithm 1 HJDSVD and RHJDSVD methods with the fixed target τ.

```
Input: Devices to compute $A\underline{v}$ and $A^T\underline{u}$ for arbitrary vectors $\underline{u}$ and $\underline{v}$, unit-length
starting vectors $u_0$ and $v_0$, the target $\tau$, and the convergence tolerance $tol$.
Output: A converged approximation $(\theta, u, v)$ to the desired $(\sigma, u^*, v^*)$ with $\sigma$
closest to $\tau$ satisfying
    $\|r\| = \left\| \begin{bmatrix} Av-\theta u \\ A^T u-\theta v \end{bmatrix} \right\| \leq \|A\|_1 \cdot tol.$

Initialization: $u_+ = u_0$, $v_+ = v_0$; $U = [\ ]$, $V = [\ ]$.
for $m = 1, 2, \dots$ do
    Set $U = [U, u_+]$, $V = [V, v_+]$, and update $H = U^T AV$, $G^{(1)} = U^T AA^T U$,
    $G^{(2)} = V^T A^T AV$, $F = \begin{bmatrix} -\tau I_m & H \\ H^T & -\tau I_m \end{bmatrix}$, and $G = \begin{bmatrix} G^{(1)}+\tau^2 I_m & -2\tau H \\ -2\tau H^T & G^{(2)}+\tau^2 I_m \end{bmatrix}$.
    Compute the largest eigenvalue $\frac{1}{\nu}$ in magnitude of $(F, G)$, its associated eigen-
    vector $\widetilde{f} = \left[ \widetilde{c}^T,\ \widetilde{d}^T \right]^T$, and $\rho = \frac{|\widetilde{c}^T H \widetilde{d}|}{\|\widetilde{c}\| \|\widetilde{d}\|}$.
    RHJDSVD: Form $G' = \begin{bmatrix} G^{(1)}+\rho^2 I_m & -2\rho H \\ -2\rho H^T & G^{(2)}+\rho^2 I_m \end{bmatrix}$. Compute the eigenvector $\widehat{f} =$
    $\left[ \widehat{c}^T, \widehat{d}^T \right]^T$ corresponding to the smallest eigenvalue of $G'$ and $\rho' = \frac{|\widehat{c}^T H \widehat{d}|}{\|\widehat{c}\| \|\widehat{d}\|}$.
    Compute the approximate singular triplet
        $(\theta, u, v) = \begin{cases} (\rho, \widetilde{u} = U\frac{\widetilde{c}}{\|\widetilde{c}\|}, \widetilde{v} = V\frac{\widetilde{d}}{\|\widetilde{d}\|} (\text{or} - V\frac{\widetilde{d}}{\|\widetilde{d}\|})) & \text{for HJDSVD,} \\ (\rho', \widehat{u} = U\frac{\widehat{c}}{\|\widehat{c}\|}, \widehat{v} = V\frac{\widehat{d}}{\|\widehat{d}\|} (\text{or} - V\frac{\widehat{d}}{\|\widehat{d}\|})) & \text{for RHJDSVD,} \end{cases}$
    as well as the residual $r = \begin{bmatrix} Av-\theta u \\ A^T u-\theta v \end{bmatrix}$.
    if $\|r\| \leq \|A\|_1 \cdot tol$ then return $(\theta, u, v)$ and stop.
    Solve the correction equation
        $\begin{bmatrix} I_M - uu^T & \\ & I_N - vv^T \end{bmatrix} \begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix} \begin{bmatrix} I_M - uu^T & \\ & I_N - vv^T \end{bmatrix} \begin{bmatrix} s \\ t \end{bmatrix} = -r,$
    where $(s, t) \perp\perp (u, v)$.
    Orthonormalize $s$ and $t$ against $U$ and $V$ to get $u_+$ and $v_+$.
end for
```

Making use of the fact that $\begin{bmatrix} I_M - uu^T & \\ & I_N - vv^T \end{bmatrix} \begin{bmatrix} s \\ t \end{bmatrix} = \begin{bmatrix} s \\ t \end{bmatrix}$, we transform (2.2) into

$$
\begin{aligned}
\begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix} \begin{bmatrix} s \\ t \end{bmatrix} &= \begin{bmatrix} uu^T & \\ & vv^T \end{bmatrix} \begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix} \begin{bmatrix} s \\ t \end{bmatrix} - \begin{bmatrix} Av - \theta u \\ A^T u - \theta v \end{bmatrix} \\
&= \begin{bmatrix} (u^T At - \tau u^T s)u \\ (v^T A^T s - \tau v^T t)v \end{bmatrix} - \begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} + \begin{bmatrix} (\theta - \tau)u \\ (\theta - \tau)v \end{bmatrix} \\
&= - \begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} + \begin{bmatrix} (\theta - \tau + u^T At)u \\ (\theta - \tau + v^T A^T s)v \end{bmatrix} \qquad (3.1)
\end{aligned}
$$

with the last equation holding since $(s, t) \perp\perp (u, v)$ means $u^T s = v^T t = 0$.

Assume that $\tau > 0$ is not a singular value of A. Since the eigenvalues of $\begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix}$ are $\pm\sigma_i - \tau,\ i = 1, 2, \dots, N$ and $-\tau$, $\begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix}$ is nonsingular. Introduce the matrix

$$B = \begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix}^{-1}. \tag{3.2}$$

Then (3.1) means that the exact solution to (2.2) is

$$\begin{bmatrix} s \\ t \end{bmatrix} = -\begin{bmatrix} u \\ v \end{bmatrix} + B \begin{bmatrix} \alpha u \\ \beta v \end{bmatrix} \quad \text{with} \quad \begin{cases} \alpha = \theta - \tau + u^T A t, \\ \beta = \theta - \tau + v^T A^T s. \end{cases} \tag{3.3}$$

Let $(\widetilde{s}, \widetilde{t})$ be an approximate solution of (2.2), and define its relative error by

$$\varepsilon = \frac{\left\| \begin{bmatrix} \widetilde{s} \\ \widetilde{t} \end{bmatrix} - \begin{bmatrix} s \\ t \end{bmatrix} \right\|}{\sqrt{\|s\|^2 + \|t\|^2}}. \tag{3.4}$$

Then we can write

$$\begin{bmatrix} \widetilde{s} \\ \widetilde{t} \end{bmatrix} = \begin{bmatrix} s \\ t \end{bmatrix} + \varepsilon \left\| \begin{bmatrix} s \\ t \end{bmatrix} \right\| \begin{bmatrix} g \\ h \end{bmatrix},$$

where $[g^T, h^T]^T$ is the normalized error direction vector with $\|g\|^2 + \|h\|^2 = 1$. As a result, we obtain

$$\begin{bmatrix} (I_M - P_U)\widetilde{s} \\ (I_N - P_V)\widetilde{t} \end{bmatrix} = \begin{bmatrix} (I_M - P_U)s \\ (I_N - P_V)t \end{bmatrix} + \varepsilon \left\| \begin{bmatrix} s \\ t \end{bmatrix} \right\| \begin{bmatrix} g_\perp \\ h_\perp \end{bmatrix}, \tag{3.5}$$

where $g_\perp = (I_M - P_U)g$ and $h_\perp = (I_N - P_V)h$. Define

$$\widetilde{u}_+ = \frac{(I_M - P_U)\widetilde{s}}{\|(I_M - P_U)\widetilde{s}\|} \quad \text{and} \quad \widetilde{v}_+ = \frac{(I_N - P_V)\widetilde{t}}{\|(I_N - P_V)\widetilde{t}\|}, \tag{3.6}$$

which are the normalized expansion vectors used to expand the current subspaces $\mathcal{U}$ and $\mathcal{V}$, respectively.

We measure the differences between $(I_M - P_U)\widetilde{s}$ and $(I_M - P_U)s$ and between $(I_N - P_V)\widetilde{t}$ and $(I_N - P_V)t$ by the relative errors

$$\widetilde{\varepsilon}_s = \frac{\|(I_M - P_U)\widetilde{s} - (I_M - P_U)s\|}{\|(I_M - P_U)s\|}, \tag{3.7}$$

$$\widetilde{\varepsilon}_t = \frac{\|(I_N - P_V)\widetilde{t} - (I_N - P_V)t\|}{\|(I_N - P_V)t\|}, \tag{3.8}$$

respectively, or, equivalently, by the sines $\sin\angle(\widetilde{u}_+, u_+)$ and $\sin\angle(\widetilde{v}_+, v_+)$. As a matter of fact, from [12, Lemma 1], it holds that

$$\begin{aligned} \sin\angle(\widetilde{u}_+, u_+) &= \widetilde{\varepsilon}_s \sin\angle(\widetilde{u}_+, g_\perp), \\ \sin\angle(\widetilde{v}_+, v_+) &= \widetilde{\varepsilon}_t \sin\angle(\widetilde{v}_+, h_\perp). \end{aligned}$$

Note that $\sin\angle(\widetilde{u}_+, g_\perp)$ and $\sin\angle(\widetilde{v}_+, h_\perp)$ are generally modest since $\widetilde{u}_+$ and $\widetilde{v}_+$ lie in the orthogonal complements of $\mathcal{U}$ and $\mathcal{V}$, respectively, and $g_\perp$ and $h_\perp$ are general vectors in the corresponding orthogonal complements. The above two relations show that $\sin\angle(\widetilde{u}_+, u_+)$ and $\sin\angle(\widetilde{v}_+, v_+)$ play the same role as $\widetilde{\varepsilon}_s$ and $\widetilde{\varepsilon}_t$ when measuring the differences between the inexact and exact expansion vectors.

Denote $\widetilde{U}_+ = [U, u_+]$ and $\widetilde{V}_+ = [V, v_+]$, whose columns form orthonormal bases of $\widetilde{\mathcal{U}}_+ = span\{\widetilde{U}_+\}$ and $\widetilde{\mathcal{V}}_+ = span\{\widetilde{V}_+\}$, respectively. In order to make the inexact JDSVD methods mimic the exact JDSVD methods well, a *necessary* condition is that the two pairs of expanded subspaces $\widetilde{\mathcal{U}}_+$, $\widetilde{\mathcal{V}}_+$ and $\mathcal{U}_+$, $\mathcal{V}_+$ have very comparable qualities, namely, $\sin\angle(\widetilde{\mathcal{U}}_+, u^*) \approx \sin\angle(\mathcal{U}_+, u^*)$ and $\sin\angle(\widetilde{\mathcal{V}}_+, v^*) \approx \sin\angle(\mathcal{V}_+, v^*)$. This condition is also *sufficient* for the inexact and exact RHJDSVD, though it is not for the inexact and exact HJDSVD [13, 27]. The following lemma is adapted from [12, Theorem 4.1].

LEMMA 3.1. *Assume that* $\sin\angle(u_+, u_\perp^*) \neq 0$ *and* $\sin\angle(v_+, v_\perp^*) \neq 0$ *with* $u_\perp^* = (I_M - P_U)u^*$ *and* $v_\perp^* = (I_N - P_V)v^*$,[1] *and let* $\widetilde{\varepsilon}_s$ *and* $\widetilde{\varepsilon}_t$ *be defined by* (3.7) *and* (3.8). *Then we have*

$$\begin{aligned} \sin\angle(\mathcal{U}_+, u^*) &= \sin\angle(\mathcal{U}, u^*)\sin\angle(u_+, u_\perp^*), \\ \sin\angle(\mathcal{V}_+, v^*) &= \sin\angle(\mathcal{V}, v^*)\sin\angle(v_+, v_\perp^*), \end{aligned} \tag{3.9}$$

and

$$\frac{\sin\angle(\widetilde{\mathcal{U}}_+, u^*)}{\sin\angle(\mathcal{U}_+, u^*)} = \frac{\sin\angle(\widetilde{u}_+, u_\perp^*)}{\sin\angle(u_+, u_\perp^*)}, \qquad \frac{\sin\angle(\widetilde{\mathcal{V}}_+, v^*)}{\sin\angle(\mathcal{V}_+, v^*)} = \frac{\sin\angle(\widetilde{v}_+, v_\perp^*)}{\sin\angle(v_+, v_\perp^*)}. \tag{3.10}$$

If $\tau_s = \frac{2\widetilde{\varepsilon}_s}{\sin\angle(u_+, u_\perp^*)} < 1$ *and* $\tau_t = \frac{2\widetilde{\varepsilon}_t}{\sin\angle(v_+, v_\perp^*)} < 1$, *we have*

$$\begin{aligned} 1 - \tau_s \leq \frac{\sin\angle(\widetilde{\mathcal{U}}_+, u^*)}{\sin\angle(\mathcal{U}_+, u^*)} \leq 1 + \tau_s, \\ 1 - \tau_t \leq \frac{\sin\angle(\widetilde{\mathcal{V}}_+, v^*)}{\sin\angle(\mathcal{V}_+, v^*)} \leq 1 + \tau_t. \end{aligned} \tag{3.11}$$

Equation (3.9) indicates that $\sin\angle(u_+, u_\perp^*)$ and $\sin\angle(v_+, v_\perp^*)$ are precisely the decreasing factors of one step subspace improvements when $\mathcal{U}$ and $\mathcal{V}$ are expanded to $\mathcal{U}_+$ and $\mathcal{V}_+$, respectively. Equation (3.10) establishes the accuracy relationships between the exact and inexact expanded subspaces. Equation (3.11) shows that, in order to make $\sin\angle(\widetilde{\mathcal{U}}_+, u^*) \approx \sin\angle(\mathcal{U}_+, u^*)$ and $\sin\angle(\widetilde{\mathcal{V}}_+, v^*) \approx \sin\angle(\mathcal{V}_+, v^*)$, τ_s and τ_t should be considerably smaller than one. Clearly, fairly small τ_s and τ_t, e.g., $\tau_s, \tau_t = 0.001$ or 0.01, are enough since we will have

$$\frac{\sin\angle(\widetilde{\mathcal{U}}_+, u^*)}{\sin\angle(\mathcal{U}_+, u^*)}, \ \frac{\sin\angle(\widetilde{\mathcal{V}}_+, v^*)}{\sin\angle(\mathcal{V}_+, v^*)} \in [0.999, 1.001] \text{ or } [0.99, 1.01] \tag{3.12}$$

and the differences between the lower and upper bounds are marginal, which means that $\widetilde{\mathcal{U}}_+$ and $\widetilde{\mathcal{V}}_+$ have essentially the same quality as $\mathcal{U}_+$ and $\mathcal{V}_+$. In the sense of (3.12), we claim that $\sin\angle(\widetilde{\mathcal{U}}_+, u^*) \approx \sin\angle(\mathcal{U}_+, u^*)$ and $\sin\angle(\widetilde{\mathcal{V}}_+, v^*) \approx \sin\angle(\mathcal{V}_+, v^*)$.

From the definition of τ_s and τ_t, we have

$$\widetilde{\varepsilon}_s = \frac{\tau_s}{2}\sin\angle(u_+, u_\perp^*), \qquad \widetilde{\varepsilon}_t = \frac{\tau_t}{2}\sin\angle(v_+, v_\perp^*). \tag{3.13}$$

Notice that $\sin\angle(u_+, u_\perp^*)$ and $\sin\angle(v_+, v_\perp^*)$ are unknown during computation. For symmetric matrices, the analysis of [12] has shown that the sizes of $\sin\angle(u_+, u_\perp^*)$

[1] If either condition fails to hold, it is seen from (3.9) that $\sin\angle(\mathcal{U}_+, u^*) = 0$ or $\sin\angle(\mathcal{V}_+, v^*) = 0$. In this case, $\mathcal{U}_+$ contains u^* or $\mathcal{V}_+$ contains v^*, that is, $\mathcal{U}_+$ or $\mathcal{V}_+$ is already the best, and the expansion of the other one subspace still satisfies the lemma.

and $\sin\angle(v_+, v_\perp^*)$ uniquely depend on the eigenvalue distribution of B, and the gap of the desired singular value σ^* and the other singular values of A: the better σ^* is separated from the others, the smaller these two quantities, so that, by (3.9), the more effectively the subspaces are expanded and the faster the JDSVD type methods converge.

Generally speaking, we should not expect that a practical SVD problem is too well conditioned, that is, $\sin\angle(u_+, u_\perp^*)$ and $\sin\angle(v_+, v_\perp^*)$ are not very small. In applications we may well assume that $\sin\angle(u_+, u_\perp^*)$ and $\sin\angle(v_+, v_\perp^*)$ are no smaller than 0.2. We should be aware that the value 0.2 means that σ^* is quite well separated from the other singular values of A. Indeed, it is instructive to see that $0.2^{15} \approx 3.3 \times 10^{-11}$, which is small enough and, by Lemma 3.1, means that after *only* 15 outer iterations the subspaces $\mathcal{U}$ and $\mathcal{V}$ already contain sufficiently accurate approximations to the desired singular vectors u^* and v^* since, at this time, $\sin\angle(\mathcal{U}, u^*) \leq 0.2^{15} \times \sin\angle(u_0, u^*)$ and $\sin\angle(\mathcal{V}, v^*) \leq 0.2^{15} \times \sin\angle(v_0, v^*)$, with u_0 and v_0 being starting vectors, are small enough. As we have elaborated, fairly small $\tau_s, \tau_t \in [10^{-3}, 10^{-2}]$ will make the inexact and exact expanded subspaces have the same quality, which should generally make the inexact JDSVD methods behave like their exact counterparts, especially for RHJDSVD, independent of the separation of σ^* from the others. As a result, suppose that $\sin\angle(u_+, u_\perp^*)$ and $\sin\angle(v_+, v_\perp^*)$ are no smaller than 0.2. In order to make the inexact JDSVD methods mimic their exact counterparts well, by (3.12) and (3.13) it is enough to take

$$\widetilde{\varepsilon}_s, \widetilde{\varepsilon}_t \in [10^{-4}, 10^{-3}] \tag{3.14}$$

when σ^* is well separated from the other singular values of A.

On the other hand, a very important point is that if $\sin\angle(u_+, u_\perp^*)$ and $\sin\angle(v_+, v_\perp^*)$ are not small, that is, the SVD problem is not so well conditioned, then JDSVD type methods may need many outer iterations to generate sufficiently accurate subspaces $\mathcal{U}$ and $\mathcal{V}$. In this case, in order to make the inexact JDSVD methods mimic their exact counterparts well, by (3.12) and (3.13) we can *relax* $\widetilde{\varepsilon}_s$ and $\widetilde{\varepsilon}_t$ and take them bigger than in (3.14). That is, if σ^* is more poorly separated from the others, $\widetilde{s}$ and $\widetilde{t}$ are allowed to have lower accuracy as approximations to s and t. In what follows, we denote

$$\widetilde{\varepsilon} = \max\{\widetilde{\varepsilon}_s, \widetilde{\varepsilon}_t\}. \tag{3.15}$$

The above analysis shows that the selection

$$\widetilde{\varepsilon} \in [10^{-4}, 10^{-3}] \tag{3.16}$$

is generally robust and reliable for the inexact JDSVD to mimic the exact JDSVD.

Our next goal is to derive relationships between the accuracy requirement ε, defined by (3.4), of the approximate solution of the correction equation (2.2) and the accuracy $\widetilde{\varepsilon}$ of the expansion vectors $\widetilde{s}$ and $\widetilde{t}$. Define

$$\widehat{\varepsilon} = \frac{\left\| \begin{bmatrix} (I_M - P_U)\widetilde{s} \\ (I_N - P_V)\widetilde{t} \end{bmatrix} - \begin{bmatrix} (I_M - P_U)s \\ (I_N - P_V)t \end{bmatrix} \right\|}{\left\| \begin{bmatrix} (I_M - P_U)s \\ (I_N - P_V)t \end{bmatrix} \right\|}, \tag{3.17}$$

which is the relative error of the *whole* inexact expansion vector $[((I_M - P_U)\widetilde{s})^T, ((I_N - P_V)\widetilde{t})^T]^T$ and the exact $[((I_M - P_U)s)^T, ((I_N - P_V)t)^T]^T$. We have the following result.

Lemma 3.2. *With* $\widetilde{\varepsilon}$ *and* $\widehat{\varepsilon}$ *defined by* (3.15) *and* (3.17), *it holds that*

$$\widehat{\varepsilon} \le \widetilde{\varepsilon}. \tag{3.18}$$

Proof. From (3.7), (3.8), and (3.15) we obtain

$$\begin{aligned}\widehat{\varepsilon} &= \sqrt{\frac{\|(I_M-P_U)(\widetilde{s}-s)\|^2+\|(I_N-P_V)(\widetilde{t}-t)\|^2}{\|(I_M-P_U)s\|^2+\|(I_N-P_V)t\|^2}}\\ &= \sqrt{\frac{\widetilde{\varepsilon}_s^2\|(I_M-P_U)s\|^2+\widetilde{\varepsilon}_t^2\|(I_N-P_V)t\|^2}{\|(I_M-P_U)s\|^2+\|(I_N-P_V)t\|^2}}\\ &\le \widetilde{\varepsilon}.\end{aligned}$$

□

With the help of this lemma, we can establish the first result on the relationship between the solution accuracy ε of correction equation (2.2) and $\widetilde{\varepsilon}$.

Theorem 3.3. *Let* (u,v) *be the current approximation to* (u^*,v^*), ε *and* $\widetilde{\varepsilon}$ *be defined by* (3.4) *and* (3.15), *and* α *and* β *be defined by* (3.3). *Then*

$$\varepsilon \le \frac{2\sqrt{\alpha^2+\beta^2}\,\sin_{\max}}{|\sigma-\tau|\,\left\|B\left[\begin{smallmatrix}\alpha u\\ \beta v\end{smallmatrix}\right]-\left[\begin{smallmatrix}u\\ v\end{smallmatrix}\right]\right\|\,\sqrt{\|g_\perp\|^2+\|h_\perp\|^2}}\widetilde{\varepsilon}, \tag{3.19}$$

where

$$\sin_{\max} = \max\{\sin\varphi,\sin\psi\} \tag{3.20}$$

with the acute angles

$$\varphi = \angle(u,u^*), \qquad \psi = \angle(v,v^*).$$

Proof. From (3.5) and (3.17), we obtain

$$\varepsilon = \frac{\left\|\left[\begin{smallmatrix}(I_M-P_U)\widetilde{s}\\ (I_N-P_V)\widetilde{t}\end{smallmatrix}\right]-\left[\begin{smallmatrix}(I_M-P_U)s\\ (I_N-P_V)t\end{smallmatrix}\right]\right\|}{\left\|\left[\begin{smallmatrix}s\\ t\end{smallmatrix}\right]\right\|\sqrt{\|g_\perp\|^2+\|h_\perp\|^2}} = \frac{\left\|\left[\begin{smallmatrix}(I_M-P_U) & \\ & (I_N-P_V)\end{smallmatrix}\right]\left[\begin{smallmatrix}s\\ t\end{smallmatrix}\right]\right\|}{\left\|\left[\begin{smallmatrix}s\\ t\end{smallmatrix}\right]\right\|\sqrt{\|g_\perp\|^2+\|h_\perp\|^2}}\widehat{\varepsilon}.$$

By (3.3), substituting $\left[\begin{smallmatrix}s\\ t\end{smallmatrix}\right] = -\left[\begin{smallmatrix}u\\ v\end{smallmatrix}\right]+B\left[\begin{smallmatrix}\alpha u\\ \beta v\end{smallmatrix}\right]$ into the above and making use of $u\in\mathcal{U}$ and $v\in\mathcal{V}$, we obtain $(I_M-P_U)u=0,\ (I_N-P_V)v=0$ and

$$\varepsilon = \frac{\left\|\left[\begin{smallmatrix}(I_M-P_U) & \\ & (I_N-P_V)\end{smallmatrix}\right]B\left[\begin{smallmatrix}\alpha u\\ \beta v\end{smallmatrix}\right]\right\|}{\left\|B\left[\begin{smallmatrix}\alpha u\\ \beta v\end{smallmatrix}\right]-\left[\begin{smallmatrix}u\\ v\end{smallmatrix}\right]\right\|\sqrt{\|g_\perp\|^2+\|h_\perp\|^2}}\widehat{\varepsilon}. \tag{3.21}$$

Decompose u and v into the orthogonal direct sums

$$\begin{cases}u = u^*\cos\varphi+p\sin\varphi,\\ v = v^*\cos\psi+q\sin\psi\end{cases} \tag{3.22}$$

with $p\perp u^*,\ q\perp v^*$ and $\|p\|=\|q\|=1$.

For $\sigma\ne\tau$, there exist unique α' and β' that satisfy the 2×2 linear system

$$\begin{bmatrix}-\tau & \sigma\\ \sigma & -\tau\end{bmatrix}\begin{bmatrix}\alpha'\\ \beta'\end{bmatrix} = \begin{bmatrix}\alpha\cos\varphi\\ \beta\cos\psi\end{bmatrix}, \quad \text{i.e.,} \quad \begin{bmatrix}\alpha'\\ \beta'\end{bmatrix} = \begin{bmatrix}-\tau & \sigma\\ \sigma & -\tau\end{bmatrix}^{-1}\begin{bmatrix}\alpha\cos\varphi\\ \beta\cos\psi\end{bmatrix}. \tag{3.23}$$

Since (σ, u^*, v^*) is a singular triplet of A, it follows from (3.23) and the definition (3.2) of B that

$$\begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix} \begin{bmatrix} \alpha' u^* \\ \beta' v^* \end{bmatrix} = \begin{bmatrix} \alpha \cos\varphi u^* \\ \beta \cos\psi v^* \end{bmatrix}, \quad \text{i.e.,} \quad \begin{bmatrix} \alpha' u^* \\ \beta' v^* \end{bmatrix} = B \begin{bmatrix} \alpha \cos\varphi u^* \\ \beta \cos\psi v^* \end{bmatrix}.$$

Therefore, from (3.22) and the above relation we obtain

$$B \begin{bmatrix} \alpha u \\ \beta v \end{bmatrix} = B \begin{bmatrix} \alpha \cos\varphi u^* \\ \beta \cos\psi v^* \end{bmatrix} + B \begin{bmatrix} \alpha \sin\varphi p \\ \beta \sin\psi q \end{bmatrix} = \begin{bmatrix} \alpha' u^* \\ \beta' v^* \end{bmatrix} + B \begin{bmatrix} \alpha \sin\varphi p \\ \beta \sin\psi q \end{bmatrix}. \tag{3.24}$$

Taking norms on both sides of the second relation in (3.23), we obtain

$$\sqrt{(\alpha')^2 + (\beta')^2} \le \frac{1}{|\sigma - \tau|} \sqrt{(\alpha \cos\varphi)^2 + (\beta \cos\psi)^2} \le \frac{1}{|\sigma - \tau|} \sqrt{\alpha^2 + \beta^2}. \tag{3.25}$$

For the numerator of (3.21), exploiting (3.24), we have

$$\begin{aligned} &\left\| \begin{bmatrix} I_M - P_U & \\ & I_N - P_V \end{bmatrix} B \begin{bmatrix} \alpha u \\ \beta v \end{bmatrix} \right\| \\ &\quad = \left\| \begin{bmatrix} \alpha'(I_M - P_U) u^* \\ \beta'(I_N - P_V) v^* \end{bmatrix} + \begin{bmatrix} I_M - P_U & \\ & I_N - P_V \end{bmatrix} B \begin{bmatrix} \alpha \sin\varphi p \\ \beta \sin\psi q \end{bmatrix} \right\| \\ &\quad \le \left\| \begin{bmatrix} \alpha'(I_M - P_U) u^* \\ \beta'(I_N - P_V) v^* \end{bmatrix} \right\| + \left\| \begin{bmatrix} I_M - P_U & \\ & I_N - P_V \end{bmatrix} B \begin{bmatrix} \alpha \sin\varphi p \\ \beta \sin\psi q \end{bmatrix} \right\| \\ &\quad \le \sqrt{(\alpha')^2 \|u_\perp^*\|^2 + (\beta')^2 \|v_\perp^*\|^2} + \frac{1}{|\sigma - \tau|} \sqrt{\alpha^2 \sin^2\varphi + \beta^2 \sin^2\psi}, \end{aligned} \tag{3.26}$$

where we have used $\|B\| = \frac{1}{|\sigma - \tau|}$ in the last inequality and $u_\perp^* = (I_M - P_U) u^*$, $v_\perp^* = (I_N - P_V) v^*$, as defined in Lemma 3.1. Notice from (3.20) that

$$\begin{aligned} \|u_\perp^*\| &= \sin\angle(\mathcal{U}, u^*) \le \sin\angle(u, u^*) = \sin\varphi \le \sin_{\max}, \\ \|v_\perp^*\| &= \sin\angle(\mathcal{V}, v^*) \le \sin\angle(v, v^*) = \sin\psi \le \sin_{\max}. \end{aligned}$$

Therefore, from the above relations and (3.25) we obtain

$$\begin{aligned} \left\| \begin{bmatrix} I_M - P_U & \\ & I_N - P_V \end{bmatrix} B \begin{bmatrix} \alpha u \\ \beta v \end{bmatrix} \right\| &\le \left(\sqrt{(\alpha')^2 + (\beta')^2} + \frac{1}{|\sigma - \tau|} \sqrt{\alpha^2 + \beta^2} \right) \sin_{\max} \\ &= \frac{2}{|\sigma - \tau|} \sqrt{\alpha^2 + \beta^2} \sin_{\max}. \end{aligned} \tag{3.27}$$

Then from (3.18) and (3.21) it follows that (3.19) holds. □

In the following we analyze Theorem 3.3 and obtain a more compact and insightful form by estimating α and β accurately. To this end, assume that the current u and v are reasonably good approximations to the desired u^* and v^* with the same order accuracy, that is,

$$\sin\varphi = \sin\angle(u, u^*) = \mathcal{O}(\epsilon), \qquad \sin\psi = \sin\angle(v, v^*) = \mathcal{O}(\epsilon) \tag{3.28}$$

with ϵ reasonably small, which means that the exact solutions $[s^T, t^T]^T$ of the correction equations in Algorithm 1 satisfy

$$\|s\| = \mathcal{O}(\epsilon), \qquad \|t\| = \mathcal{O}(\epsilon). \tag{3.29}$$

Define the quantity

$$\gamma = sign(\theta - \tau)\frac{\sqrt{2}}{\sqrt{\alpha^2 + \beta^2}} \tag{3.30}$$

with $sign(\cdot)$ being the sign function.

In what follows we estimate $\alpha,\ \beta$, and γ accurately.

THEOREM 3.4. *For α, β, and γ defined by* (3.3) *and* (3.30), *we have*

$$\alpha = \theta - \tau + \mathcal{O}(\epsilon^2), \tag{3.31}$$

$$\beta = \theta - \tau + \mathcal{O}(\epsilon^2), \tag{3.32}$$

$$\gamma = \frac{1}{\theta - \tau} + \mathcal{O}(\epsilon^2). \tag{3.33}$$

Proof. From (3.3) and (3.22), we have

$$\begin{cases} \alpha = \theta - \tau + t^T A^T (u^* \cos\varphi + p \sin\varphi), \\ \beta = \theta - \tau + s^T A (v^* \cos\psi + q \sin\psi), \end{cases}$$

i.e.,

$$\begin{cases} \alpha - (\theta - \tau) = t^T A^T p \sin\varphi + \sigma t^T v^* \cos\varphi, \\ \beta - (\theta - \tau) = s^T A q \sin\psi + \sigma s^T u^* \cos\psi. \end{cases}$$

Keep in mind that $\varphi = \angle(u, u^*), \quad \psi = \angle(v, v^*)$. Similarly to (3.22), we now decompose u^* and v^* into the orthogonal direct sums

$$\begin{cases} u^* = u \cos\varphi + p' \sin\varphi, \\ v^* = v \cos\psi + q' \sin\psi \end{cases}$$

with $p' \perp u, q' \perp v$, and $\|p'\| = \|q'\| = 1$. Since $s \perp u$ and $t \perp v$, we have

$$\begin{cases} \alpha - (\theta - \tau) = t^T A^T p \sin\varphi + \sigma t^T q' \cos\varphi \sin\psi, \\ \beta - (\theta - \tau) = s^T A q \sin\psi + \sigma s^T p' \cos\psi \sin\varphi. \end{cases}$$

Therefore, from (3.20), (3.28), and (3.29) we obtain

$$\begin{cases} |\alpha - (\theta - \tau)| \le (\|A\| + \sigma)\|t\| \sin_{\max} = \mathcal{O}(\epsilon^2), \\ |\beta - (\theta - \tau)| \le (\|A\| + \sigma)\|s\| \sin_{\max} = \mathcal{O}(\epsilon^2). \end{cases}$$

Consequently, we obtain

$$\alpha = \theta - \tau + \mathcal{O}(\epsilon^2), \qquad \beta = \theta - \tau + \mathcal{O}(\epsilon^2),$$

which proves (3.31), (3.32), and

$$|\alpha - \beta| \le (\|A\| + \sigma)(\|t\| + \|s\|) \sin_{\max} \le 2\|A\|(\|t\| + \|s\|) \sin_{\max} = \mathcal{O}(\epsilon^2). \tag{3.34}$$

From (3.31) and (3.32) it is seen that α and β have the same signs as $\theta - \tau$ for ϵ reasonably small. Moreover, α, β, and γ have the same signs, and

$$\alpha\gamma = 1 + \mathcal{O}(\epsilon^2) \quad \text{and} \quad \beta\gamma = 1 + \mathcal{O}(\epsilon^2),$$

which yields (3.33) by combining (3.30) with (3.31) or (3.32). □

By simple justification, we find from this theorem that

$$\begin{aligned}\frac{1}{\sqrt{\alpha^2+\beta^2}}\left\|B\begin{bmatrix}\alpha u\\ \beta v\end{bmatrix}-\begin{bmatrix}u\\ v\end{bmatrix}\right\| &= \frac{1}{\sqrt{2}}\left\|B\begin{bmatrix}u\\ v\end{bmatrix}-\gamma\begin{bmatrix}u\\ v\end{bmatrix}\right\|+\mathcal{O}(\epsilon^2)\\ &= \frac{1}{\sqrt{2}}\left\|B\begin{bmatrix}u\\ v\end{bmatrix}-\frac{1}{\theta-\tau}\begin{bmatrix}u\\ v\end{bmatrix}\right\|+\mathcal{O}(\epsilon^2).\end{aligned} \tag{3.35}$$

Therefore, within $\mathcal{O}(\epsilon^2)$, the above left-hand side, which is a factor of the right-hand side in (3.19), is the residual norm of regarding γ or $\frac{1}{\theta-\tau}$ as an approximation to the eigenvalue $\frac{1}{\sigma-\tau}$ of B and $\frac{1}{\sqrt{2}}\left[\begin{smallmatrix}u\\ v\end{smallmatrix}\right]$ as the corresponding normalized approximate eigenvector.

With the help of Theorem 3.4, we are able to make Theorem 3.3 clearer, more compact, and more insightful. Before doing so, we need the following lemma, which is direct from [16, Theorem 6.1].

Lemma 3.5. *Let $(\frac{1}{\sigma-\tau}, w=\frac{1}{\sqrt{2}}\left[\begin{smallmatrix}u^*\\ v^*\end{smallmatrix}\right])$ be a simple eigenpair of B and $[w, W_\perp]$ be orthogonal. Then*

$$\begin{bmatrix}w^T\\ W_\perp^T\end{bmatrix}B[w,W_\perp]=\begin{bmatrix}\frac{1}{\sigma-\tau} & 0\\ 0 & L\end{bmatrix},$$

where $L=W_\perp^T B W_\perp$ is a symmetric matrix. Let $(\gamma, z=\frac{1}{\sqrt{2}}\left[\begin{smallmatrix}u\\ v\end{smallmatrix}\right])$ be an approximation to $(\frac{1}{\sigma-\tau}, w)$. Assume that γ is not an eigenvalue of L, and define

$$\mathrm{sep}(\gamma, L)=\|(L-\gamma I)^{-1}\|^{-1}>0.$$

Then

$$\sin\angle(z,w)\le\frac{\|Bz-\gamma z\|}{\mathrm{sep}(\gamma,L)}. \tag{3.36}$$

Based on Theorem 3.3 and Lemma 3.5, we can establish the following close relationship between ε and $\widetilde{\varepsilon}$ defined by (3.4) and (3.15), respectively.

Theorem 3.6. *Assume that $\gamma = sign(\theta-\tau)\frac{\sqrt{2}}{\sqrt{\alpha^2+\beta^2}}$ is an approximation to $\frac{1}{\sigma-\tau}$ and is not an eigenvalue of L, and let ε and $\widetilde{\varepsilon}$ be defined by* (3.4) *and* (3.15)*. Then*

$$\varepsilon\le\frac{2\sqrt{2}\delta}{\mathrm{sep}(\gamma,L)|\sigma-\tau|\sqrt{\|g_\perp\|^2+\|h_\perp\|^2}}\widetilde{\varepsilon}, \tag{3.37}$$

where

$$\delta=\frac{\left\|B\left[\begin{smallmatrix}u\\ v\end{smallmatrix}\right]-\gamma\left[\begin{smallmatrix}u\\ v\end{smallmatrix}\right]\right\|}{\left\|B\left[\begin{smallmatrix}\alpha\gamma u\\ \beta\gamma v\end{smallmatrix}\right]-\gamma\left[\begin{smallmatrix}u\\ v\end{smallmatrix}\right]\right\|}. \tag{3.38}$$

Proof. Denote $\phi=\angle\left(\left[\begin{smallmatrix}u\\ v\end{smallmatrix}\right],\left[\begin{smallmatrix}u^*\\ v^*\end{smallmatrix}\right]\right)$. Then $\phi=\angle(z,w)$ by noting the definition of z and w in Lemma 3.5. From (3.36) we obtain

$$\sin\phi\le\frac{\left\|B\left[\begin{smallmatrix}u\\ v\end{smallmatrix}\right]-\gamma\left[\begin{smallmatrix}u\\ v\end{smallmatrix}\right]\right\|}{\sqrt{2}\mathrm{sep}(\gamma,L)}.$$

From [14, Theorem 2.3], we have

$$\sin^2\varphi+\sin^2\psi\le 2\sin^2\phi.$$

Hence

$$\sin_{\max} = \max\{\sin\varphi, \sin\psi\} \le \sqrt{2}\sin\phi \le \frac{\left\|B\left[\begin{smallmatrix}u\\v\end{smallmatrix}\right] - \gamma\left[\begin{smallmatrix}u\\v\end{smallmatrix}\right]\right\|}{\operatorname{sep}(\gamma, L)}. \tag{3.39}$$

Making use of this relation and δ defined by (3.38), we obtain from (3.19) that

$$\begin{aligned}\varepsilon &\le \frac{2\sqrt{2}\delta\sin_{\max}}{|\sigma-\tau|\left\|B\left[\begin{smallmatrix}u\\v\end{smallmatrix}\right] - \gamma\left[\begin{smallmatrix}u\\v\end{smallmatrix}\right]\right\|\sqrt{\|g_\perp\|^2+\|h_\perp\|^2}}\widetilde{\varepsilon}\\ &\le \frac{2\sqrt{2}\delta}{\operatorname{sep}(\gamma, L)|\sigma-\tau|\sqrt{\|g_\perp\|^2+\|h_\perp\|^2}}\widetilde{\varepsilon}.\end{aligned}$$

□

Theorem 3.6 shows that once $\widetilde{\varepsilon}$ is given we can determine the *least* or *lowest* accuracy requirement ε for the correction equation (2.2).

We will prove that $\delta = 1 + \mathcal{O}(\epsilon)$ afterward. Equation (3.5) indicates that $\sqrt{\|g_\perp\|^2+\|h_\perp\|^2}$ is modest since g and h are general vectors satisfying $\|g\|^2+\|h\|^2 = 1$ and $\|g_\perp\| \le \|g\|$, $\|h_\perp\| \le \|h\|$. If $\sqrt{\|g_\perp\|^2+\|h_\perp\|^2}$ is small, the solution accuracy requirement ε can be relaxed for a fixed small $\widetilde{\varepsilon}$, that is, the correction equation (2.2) is allowed to be solved with *lower* accuracy. So a small $\sqrt{\|g_\perp\|^2+\|h_\perp\|^2}$ is a *lucky* event.

If γ is well separated from the eigenvalues of B other than $\frac{1}{\sigma-\tau}$, then we have $\operatorname{sep}(\gamma, L) \approx \operatorname{sep}(\frac{1}{\sigma-\tau}, L) \le 2\|B\| = \frac{2}{|\sigma-\tau|}$, which is tight. In this case, noting that $\sqrt{\|g_\perp\|^2+\|h_\perp\|^2} \le 1$, we obtain

$$\varepsilon \le \frac{2\sqrt{2}\delta}{\operatorname{sep}(\gamma, L)|\sigma-\tau|\sqrt{\|g_\perp\|^2+\|h_\perp\|^2}}\widetilde{\varepsilon} = \frac{\sqrt{2}}{\mathcal{O}(1)}\widetilde{\varepsilon} = \mathcal{O}(\widetilde{\varepsilon}).$$

On the other hand, if $\frac{1}{\sigma-\tau}$ is not well separated from the other eigenvalues of B, then $\operatorname{sep}(\gamma, L) \approx \operatorname{sep}(\frac{1}{\sigma-\tau}, L)$ is considerably smaller than $\|B\| = \frac{1}{|\sigma-\tau|}$, which may make ε considerably bigger than $\widetilde{\varepsilon}$, meaning that we are allowed to solve (2.2) with *lower* accuracy. Therefore, as far as solving correction equation (2.2) is concerned, the bad separation of $\frac{1}{\sigma-\tau}$ from the other eigenvalues of B is a lucky event.

Summarizing the above, we conclude that, in any event, $\varepsilon = c\widetilde{\varepsilon}$ with a fairly modest constant $c = \mathcal{O}(1)$ is a reliable and general-purpose choice.

Now, we analyze δ and estimate its size accurately. Recall (3.30), and denote

$$x = B\begin{bmatrix}\alpha u\\ \beta v\end{bmatrix} - \begin{bmatrix}u\\v\end{bmatrix}, \qquad y = \frac{1}{\gamma}\left(B\begin{bmatrix}u\\v\end{bmatrix} - \gamma\begin{bmatrix}u\\v\end{bmatrix}\right). \tag{3.40}$$

Then it is known from (3.38) that $\delta = \frac{\|y\|}{\|x\|}$ and

$$\left|1 - \frac{1}{\delta}\right| = \frac{|\|y\| - \|x\||}{\|y\|} \le \frac{\|y-x\|}{\|y\|}. \tag{3.41}$$

When the right-hand side of (3.41) is smaller than one, we will have

$$\frac{1}{1+\frac{\|y-x\|}{\|y\|}} \le \delta \le \frac{1}{1-\frac{\|y-x\|}{\|y\|}}. \tag{3.42}$$

We next estimate the right-hand side of (3.41) and prove that it is $\mathcal{O}(\epsilon)$.

Write $z = \frac{1}{\sqrt{2}}\left[\begin{smallmatrix} u \\ v \end{smallmatrix}\right]$ and $w = \frac{1}{\sqrt{2}}\left[\begin{smallmatrix} u^* \\ v^* \end{smallmatrix}\right]$. Then $y = \frac{\sqrt{2}}{\gamma}(Bz - \gamma z)$, and from (3.36) we have

$$\|y\| = \frac{\sqrt{2}}{|\gamma|}\|Bz - \gamma z\| \geq \frac{\sqrt{2}}{|\gamma|} \sin\angle(z, w)\mathrm{sep}(\gamma, L).$$

Notice from (3.39) that $\sqrt{2}\sin\angle(z,w) \geq \sin_{\max}$. Then the above relation gives

$$\|y\| \geq \frac{\mathrm{sep}(\gamma, L)\sin_{\max}}{|\gamma|}. \tag{3.43}$$

As for $\|y - x\|$, from (3.30) we obtain

$$\begin{aligned}
\|y - x\| &= \left\| B \begin{bmatrix} (\frac{1}{\gamma} - \alpha)u \\ (\frac{1}{\gamma} - \beta)v \end{bmatrix} \right\| \leq \|B\| \sqrt{(\frac{1}{\gamma} - \alpha)^2 + (\frac{1}{\gamma} - \beta)^2} \\
&= \frac{1}{|\sigma - \tau|} \sqrt{2(\alpha^2 + \beta^2) - \sqrt{2(\alpha^2 + \beta^2)}\mathrm{sign}(\theta - \tau)(\alpha + \beta)} \\
&= \frac{1}{|\sigma - \tau|} \sqrt{\sqrt{2(\alpha^2 + \beta^2)}[\sqrt{2(\alpha^2 + \beta^2)} - \mathrm{sign}(\theta - \tau)(\alpha + \beta)]} \\
&= \frac{1}{|\sigma - \tau|} \sqrt{\frac{\sqrt{2\alpha^2 + 2\beta^2}(\alpha - \beta)^2}{\sqrt{2\alpha^2 + 2\beta^2} + \mathrm{sign}(\theta - \tau)(\alpha + \beta)}}.
\end{aligned}$$

It is known from Theorem 3.4 that $sign(\alpha) = sign(\beta) = sign(\theta - \tau)$ for ϵ reasonably small. Consequently, we obtain $sign(\theta - \tau)(\alpha + \beta) > 0$. Therefore,

$$\|y - x\| \leq \frac{1}{|\sigma - \tau|}|\alpha - \beta|. \tag{3.45}$$

From (3.34), (3.43), and (3.45) we obtain

$$\frac{\|y - x\|}{\|y\|} \leq \frac{|\gamma||\alpha - \beta|}{|\sigma - \tau| \sin_{\max} \mathrm{sep}(\gamma, L)} \leq \frac{2|\gamma|\|A\|(\|s\| + \|t\|)}{|\sigma - \tau|\mathrm{sep}(\gamma, L)} = \mathcal{O}(\epsilon). \tag{3.46}$$

Summarizing the above derivation, we have established the following results.

THEOREM 3.7. *Assume that* $\sin\varphi = \mathcal{O}(\epsilon)$, $\sin\psi = \mathcal{O}(\epsilon)$, $\|s\| = \mathcal{O}(\epsilon)$, *and* $\|t\| = \mathcal{O}(\epsilon)$. *If* ϵ *is sufficiently small such that* $\|s\| + \|t\| < \frac{|\sigma - \tau|\mathrm{sep}(\gamma, L)}{2|\gamma|\|A\|}$, *we have*

$$\frac{1}{1 + \frac{2|\gamma|\|A\|(\|s\| + \|t\|)}{|\sigma - \tau|\mathrm{sep}(\gamma, L)}} \leq \delta \leq \frac{1}{1 - \frac{2|\gamma|\|A\|(\|s\| + \|t\|)}{|\sigma - \tau|\mathrm{sep}(\gamma, L)}}, \tag{3.47}$$

i.e.,

$$1 - \mathcal{O}(\epsilon) \leq \delta \leq 1 + \mathcal{O}(\epsilon). \tag{3.48}$$

Proof. Combining (3.42) and (3.46) gives

$$\left|1 - \frac{1}{\delta}\right| \leq \frac{2|\gamma|\|A\|(\|s\| + \|t\|)}{|\sigma - \tau|\mathrm{sep}(\gamma, L)} = \mathcal{O}(\epsilon),$$

which means (3.47) and (3.48). □

This theorem indicates that, in practical implementations, we can take $\delta = 1$ when determining ε from a given $\widetilde{\varepsilon}$.

4. JDSVD type methods with deflation.

Suppose that the ℓ singular values σ_i closest to the target τ and the associated singular vectors u_i and v_i are required, $i = 1, 2, \dots, \ell$. We assume that the σ_i satisfy

$$|\sigma_1 - \tau| < |\sigma_2 - \tau| < \cdots < |\sigma_\ell - \tau| < |\sigma_{\ell+1} - \tau| \leq \cdots \leq |\sigma_N - \tau|. \tag{4.1}$$

Our JDSVD type methods described in section 2 combined with some deflation technique can meet the demands, and the theoretical results established in section 3 can be extended to the variants.

In this section, we assume that $k < \ell$ approximate singular values and the corresponding approximate singular vectors written as

$$(\Theta_c, U_c, V_c) = \left(\begin{bmatrix} \theta_{(1,c)} & & \\ & \ddots & \\ & & \theta_{(k,c)} \end{bmatrix}, [u_{(1,c)}, \dots, u_{(k,c)}], [v_{(1,c)}, \dots, v_{(k,c)}] \right) \tag{4.2}$$

have already converged to the desired singular triplets $(\sigma_i, u_i, v_i),\ i = 1, 2, \dots, k$ with the σ_i labeled as (4.1), that is, the residuals of the approximate singular triplets satisfy

$$\|r_{(i,c)}\| = \left\| \begin{bmatrix} Av_{(i,c)} - \theta_{(i,c)} u_{(i,c)} \\ A^T u_{(i,c)} - \theta_{(i,c)} v_{(i,c)} \end{bmatrix} \right\| \leq \|A\|_1 \cdot tol, \qquad i = 1, \dots, k, \tag{4.3}$$

where *tol* is the user prescribed tolerance. The columns of U_c and V_c are orthogonal and (Θ_c, U_c, V_c) is the approximation to the partial SVD of A:

$$(\Sigma_k, U_k, V_k) = \left(\begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_k \end{bmatrix}, [u_1, \dots, u_k], [v_1, \dots, v_k] \right). \tag{4.4}$$

Our goal is to compute the next singular triplet $(\sigma_{k+1}, u_{k+1}, v_{k+1})$, which is denoted as (σ, u^*, v^*) for simplicity, as done in section 1.

4.1. Deflation.

The deflation applied to our JDSVD type methods works as follows. Given a pair of m-dimensional searching subspaces $\mathcal{U}$ and $\mathcal{V}$ orthogonal to U_c and V_c, we use each of the extraction methods described in sections 2.1–2.2 to extract an approximate singular triplet (θ, u, v) satisfying the double orthogonality condition, whose residual $r = r(\theta, u, v)$ is defined by (2.1). When expanding the subspaces, different from (2.2), we instead solve a new correction equation of form

$$\begin{bmatrix} I_M - QQ^T & \\ & I_N - ZZ^T \end{bmatrix} \begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix} \begin{bmatrix} I_M - QQ^T & \\ & I_N - ZZ^T \end{bmatrix} \begin{bmatrix} s \\ t \end{bmatrix} = -r \tag{4.5}$$

for $(s, t) \perp\perp (Q, Z)$, where the columns of $Q = [U_c, u]$ and $Z = [V_c, v]$ are orthonormal. Then we orthonormalize s and t against U and V, whose columns are orthonormal bases of $\mathcal{U}$ and $\mathcal{V}$, respectively, to obtain the expansion vectors u_+ and v_+. Expand U and V to $U_+ = [U, u_+]$ and $V_+ = [V, v_+]$, respectively, whose columns form orthonormal bases of the expanded searching subspaces $\mathcal{U}_+$ and $\mathcal{V}_+$, from which we continue to extract a new approximate singular triplet.

Obviously, new approximate left and right singular vectors are always orthogonal to the already converged U_c and V_c since the latter ones are orthogonal to $\mathcal{U}_+$ and $\mathcal{V}_+$, respectively. Such orthogonality is guaranteed to working precision in finite precision arithmetic, provided that s and t are orthonormalized against U and V to working precision when expanding $\mathcal{U}$ and $\mathcal{V}$ to $\mathcal{U}_+$ and $\mathcal{V}_+$, respectively.

However, we notice that, under our assumption on $\mathcal{U}$ and $\mathcal{V}$, although the current approximate singular vectors (θ, u, v) naturally satisfy $(u, v) \perp\perp (U_c, V_c)$, the relation $r(\theta, u, v) \perp\perp (U_c, V_c)$ holds only when $(U_c, V_c) = (U_k, V_k)$. In order to guarantee the consistency of the correction equation (4.5), we should use the projected residual

$$r_p = \begin{bmatrix} I_M - U_cU_c^T & \\ & I_N - V_cV_c^T \end{bmatrix} r(\theta, u, v) \tag{4.6}$$

to replace the right-hand side $r(\theta, u, v)$ in (4.5).

Notice that current m-dimensional searching subspaces $\mathcal{U}$ and $\mathcal{V}$ contain reasonable information on the next desired $(u^*, v^*) = (u_{k+1}, v_{k+1})$. Therefore, when computing $(\sigma_{k+1}, u_{k+1}, v_{k+1})$ with the current (θ, u, v) already converged to (σ_k, u_k, v_k), we may benefit a lot from $\mathcal{U}$ and $\mathcal{V}$ and find a better initial approximation to (u^*, v^*) rather than the one generated in some random way. We will purge the newly converged left and right singular vectors from $\mathcal{U}$ and $\mathcal{V}$ and obtain the new $(m-1)$-dimensional searching subspaces which are orthogonal to the converged left and right singular vectors, respectively, from which we use harmonic or refined harmonic extraction to compute approximate left and right singular vectors as the initial approximation to the desired (u^*, v^*). Then we proceed to expand the searching subspaces in the regular way as done in Algorithm 1.

Specifically we formally construct the desired $(m-1)$-dimensional left and right searching subspaces as follows: Let the columns of U and V form the orthonormal bases of $\mathcal{U}$ and $\mathcal{V}$, and the converged left and right singular vectors $u = Uc$ and $v = Vd$ with $\|c\| = \|d\| = 1$. Then we augment c and d such that $[c, C]$ and $[d, D]$ are orthogonal, in which the $m \times (m-1)$ orthonormal C and D are obtained by using Householder transformations to compute the full *factored* QR factorizations of the $m \times 1$ matrices c and d at a cost of $\mathcal{O}(m)$ flops. That is, we do not explicitly compute the Householder transformations by only storing the two normalized Householder vectors h_c and h_d when computing the full QR factorizations of c and d. In this case we can write C and D in the factored form of $C = (I - 2h_ch_c^T)(e_2, \ldots, e_m)$ and $D = (I - 2h_dh_d^T)(e_2, \ldots, e_m)$. We mention that if C and D are formed explicitly, then extra $\mathcal{O}(m^2)$ flops are needed. The orthonormal columns of $U_{new} = UC$ and $V_{new} = VD$ form bases of the desired $(m-1)$-dimensional searching subspaces $\mathcal{U}_{new}$ and $\mathcal{V}_{new}$, which are orthogonal to the converged u and v, respectively. The computations of $U_{new} = UC$ and $V_{new} = VD$ cost $\mathcal{O}(Mm^2)$ and $\mathcal{O}(Nm^2)$ flops for the explicit C and D, respectively, but it can easily be justified that they consume only $\mathcal{O}(Mm)$ and $\mathcal{O}(Nm)$ flops correspondingly when the factored C and D are exploited. Obviously, the latter way is more efficient to form $U_{new} = UC$ and $V_{new} = VD$.

Note that what we need is to form the matrices $H_{new} = U_{new}^T A V_{new}$, $G_{new}^{(1)} = U_{new}^T AA^T U_{new}$, and $G_{new}^{(2)} = V_{new}^T A^T A V_{new}$. The direct computation by definition is costly. In fact, we do not form U_{new} and V_{new} explicitly when computing these three matrices, and substantial computational savings can be achieved in the following way. Based on the expressions

$$\begin{aligned} H_{new} &= U_{new}^T A V_{new} = C^T(U^T A V)D = C^T H D, \\ G_{new}^{(1)} &= U_{new}^T AA^T U_{new} = C^T(U^T AA^T U)C = C^T G^{(1)} C, \\ G_{new}^{(2)} &= V_{new}^T A^T A V_{new} = D^T(V^T A^T A V)D = D^T G^{(2)} D, \end{aligned}$$

we compute each of them by using its final expression. Note that H, $G^{(1)}$, and $G^{(2)}$ are already available when computing the converged approximation to the kth left and right singular vector pair (u_k, v_k) of A. Therefore, the cost of forming $C^T H D$,

$C^TG^{(1)}C$, and $D^TG^{(2)}D$ is negligible, compared with that of computing $U_{new}^TAV_{new}$, $(A^TU_{new})^T(A^TU_{new})$, and $(AV_{new})^T(AV_{new})$. As a matter of fact, the total computational cost of H_{new}, $G^{(1)}$, and $G^{(2)}$ is $\mathcal{O}(m^3)$ flops if the explicit C and D are used. However, if the factored C and D are used, then the total computational cost of H_{new}, $G^{(1)}$, and $G^{(2)}$ can be further reduced to $\mathcal{O}(m^2)$ flops. For example, we compute H_{new} in the following order:

$$\begin{aligned} H_{new} &= (e_2,\ldots,e_m)^T(I-2h_ch_c^T)H(I-2h_dh_d^T)(e_2,\ldots,e_m)\\ &= (e_2,\ldots,e_m)^T(H-2h_c(h_c^TH)-2(Hh_d)h_d^T+4(h_c^THh_d)(h_ch_d^T))(e_2,\ldots,e_m). \end{aligned}$$

It is easily seen that the computation of $H-2h_c(h_c^TH)-2(Hh_d)h_d^T+4(h_c^THh_d)(h_ch_d^T)$ costs $\mathcal{O}(m^2)$ flops and H_{new} is the right bottom $(m-1)\times(m-1)$ submatrix of $(I-2h_ch_c^T)H(I-2h_dh_d^T)$.

4.2. Theoretical extensions. Assume that we have computed the first k singular triplets (σ_i, u_i, v_i) exactly, i.e.,

$$(\Theta_c, U_c, V_c) = (\Sigma_k, U_k, V_k). \tag{4.7}$$

We will prove that the theory established in section 3 works for the JDSVD methods with deflation described above. That is, as far as the correction equation (4.5) with the right-hand side (4.6) is concerned, there are the same relationships between the solution accuracy ε defined by (3.4) and the accuracy $\widetilde{\varepsilon}$ of the expansion vectors $\widetilde{s}$ and $\widetilde{t}$ defined by (3.15).

First, we need to prove that the solution $[s^T, t^T]^T$ of the correction equation (4.5) still has the form (3.3). By using the same derivation as (3.1) and noticing that both (s,t) and (u,v) are double orthogonal to (U_c, V_c), it is straightforward to justify that this is true.

With Lemma 3.2 and the expression (3.3), we can extend Theorem 3.3 to the solution accuracy ε of (4.5) and the accuracy $\widetilde{\varepsilon}$ of the expansion vectors $\widetilde{s}$ and $\widetilde{t}$. To this end, we only need to modify the proof followed slightly: When decomposing u and v into the orthogonal direct sums (3.22), we have $(p,q)\perp\perp(U_c,V_c)$ since both (u,v) and (u_*,v_*) are double orthogonal to (U_c,V_c). As a result, for the second term in the right-hand side of (3.24) we have

$$\left\|B\begin{bmatrix}\alpha\sin\varphi p\\ \beta\sin\psi q\end{bmatrix}\right\| \le \frac{1}{|\sigma-\tau|}\sqrt{\alpha^2\sin^2\varphi+\beta^2\sin^2\psi},^{2} \tag{4.8}$$

starting with which we repeat the remaining proof of Theorem 3.3 step by step and extend the theorem to the solution of (4.5).

To make Theorem 3.3 clearer, apart from Theorem 3.4, which trivially holds for α, β defined by (3.3) and γ defined by (3.30), we need the following lemma, which is a generalization of Lemma 3.5 and reduces to it when $k=0$.

LEMMA 4.1. *Let* $(\frac{1}{\sigma-\tau}, w=\frac{1}{\sqrt{2}}\left[\begin{smallmatrix}u^*\\ v^*\end{smallmatrix}\right])$ *be a simple eigenpair of* B *and* (Σ_k, U_k, V_k) *be defined as in* (4.4), *and let* $W=\left[\begin{smallmatrix}U_k & \\ & V_k\end{smallmatrix}\right]$ *and* $[w, W, W_\perp]$ *be orthogonal. Then*

$$\begin{bmatrix}w^T\\ W^T\\ W_\perp^T\end{bmatrix} B[w,W,W_\perp] = \begin{bmatrix}\frac{1}{\sigma-\tau} & & \\ & \left[\begin{smallmatrix}-\tau I_k & \Sigma_k\\ \Sigma_k & -\tau I_k\end{smallmatrix}\right]^{-1} & \\ & & L\end{bmatrix}, \tag{4.9}$$

[2] Since span$\{U_k, V_k\}$ is an invariant subspace of B, it is easy to verify that $\|Bx\|\le\frac{1}{|\sigma-\tau|}\|x\|$ for any $x\perp\perp(U_k,V_k)$.

where $L = W_\perp^T B W_\perp$. Suppose that $(\gamma, z = \frac{1}{\sqrt{2}}\left[\begin{smallmatrix} u \\ v \end{smallmatrix}\right])$ is an approximation to $(\frac{1}{\sigma-\tau}, w)$ satisfying $z \perp\perp (U_k, V_k)$ and γ is not an eigenvalue of L. Then

$$\sin\angle(z, w) \leq \frac{\|Bz - \gamma z\|}{\operatorname{sep}(\gamma, L)}. \tag{4.10}$$

Proof. Since $z \perp\perp (U_k, V_k)$, we have $W^T z = 0$ and

$$z = ww^T z + WW^T z + W_\perp W_\perp^T z = ww^T z + W_\perp W_\perp^T z. \tag{4.11}$$

By (4.9), since $W_\perp^T w = 0$ and the smallest singular value of $L - \gamma I$ is $\operatorname{sep}(\gamma, L)$, it holds that

$$\begin{aligned} \|(B - \gamma I)z\| &= \|(\frac{1}{\sigma - \tau} - \gamma)(w^T z)w + W_\perp (L - \gamma I) W_\perp^T z\| \\ &\geq \|W_\perp (L - \gamma I) W_\perp^T z\| = \|(L - \gamma I) W_\perp^T z\| \geq \operatorname{sep}(\gamma, L)\|W_\perp^T z\|. \end{aligned} \tag{4.12}$$

It follows from (4.11) that $\|W_\perp^T z\| = \|W_\perp W_\perp^T z\| = \|(I - ww^T)z\| = \sin\angle(z, w)$, which with (4.12) establishes (4.10). □

With the help of Lemma 4.1, it is straightforward to derive Theorem 3.6 for the solution accuracy ε of (4.5) and the accuracy $\widetilde{\varepsilon}$ of the expansion vectors, where δ satisfies the estimates in Theorem 3.7.

It is necessary to point out that our theoretical extensions above are established under the assumption (4.7), i.e., $tol = 0$. For the stopping tolerance $tol > 0$ of the outer iterations in (4.3), the new corresponding bounds in Theorems 3.3 and 3.6–3.7 are simply the counterparts already established for $tol = 0$ plus certain corresponding smaller terms $\mathcal{O}(tol)$'s. Their derivations are routine but tedious, and we do not give details in this paper.

5. Practical stopping criteria for inner iterations.

In this section, we use the results established in the last two sections to determine a practical ε from a given $\widetilde{\varepsilon}$ and derive practical stopping criteria for the inner iterations involved in the inexact JDSVD type algorithms. To this end, we will restore the notation used for HJDSVD and RHJDSVD in section 2 in order to treat the two inexact algorithms separately.

From Theorem 3.6, we take $\delta = 1$ in computation. Since we cannot compute $\sqrt{\|g_\perp\|^2 + \|h_\perp\|^2}$ directly, we simply replace it by its upper bound one, which makes ε as small as possible, so that the inexact JDSVD methods are more reliable to mimic their exact counterparts. Also, in Theorem 3.6, we replace the unknown and desired quantity σ by the current approximate singular value, i.e., ρ for HJDSVD and ρ' for RHJDSVD.

For $\operatorname{sep}(\gamma, L)$, we see from Theorem 3.4 that $\gamma \approx \frac{1}{\alpha} \approx \frac{1}{\beta} = \frac{1}{\theta - \tau} + \mathcal{O}(\epsilon^2)$. So we use $\operatorname{sep}(\frac{1}{\rho-\tau}, L)$ and $\operatorname{sep}(\frac{1}{\rho'-\tau}, L)$ to estimate $\operatorname{sep}(\gamma, L)$ in HJDSVD and RHJDSVD, respectively. However, since L is unavailable, it is impossible to compute $\operatorname{sep}(\frac{1}{\rho-\tau}, L)$ or $\operatorname{sep}(\frac{1}{\rho'-\tau}, L)$. We can exploit the eigenvalues of the matrix pencil (F, G) to estimate $\operatorname{sep}(\frac{1}{\rho-\tau}, L)$ or $\operatorname{sep}(\frac{1}{\rho'-\tau}, L)$. Let $\frac{1}{\nu_i}$ be the eigenvalues of (F, G) in (2.7) other than its largest $\frac{1}{\nu}$ in magnitude and such that $\theta_i' = \nu_i + \tau$ are positive. Then $\theta_i' = \nu_i + \tau$ are approximate singular values of A. Therefore, we have the estimates

$$\operatorname{sep}\left(\frac{1}{\rho - \tau}, L\right) \approx \min_{\theta_i' \neq \rho} \left|\frac{1}{\rho - \tau} - \frac{1}{\theta_i' - \tau}\right| \qquad \text{for HJDSVD,}$$

$$\operatorname{sep}\left(\frac{1}{\rho' - \tau}, L\right) \approx \min_{\theta_i' \neq \rho} \left|\frac{1}{\rho' - \tau} - \frac{1}{\theta_i' - \tau}\right| \qquad \text{for RHJDSVD.}$$

Taking the equality in the bound of (3.37) with all the above estimates, we obtain

$$\varepsilon_H = 2\sqrt{2}\widetilde{\varepsilon} \max_{\nu_i \neq \nu} \frac{|\nu_i|}{|\nu_i + \tau - \rho|} \tag{5.1}$$

for HJDSVD and

$$\varepsilon_{RH} = 2\sqrt{2}\widetilde{\varepsilon} \max_{\nu_i \neq \nu} \frac{|\nu_i|}{|\nu_i + \tau - \rho'|} \tag{5.2}$$

for RHJDSVD, respectively.

Based on the above results, we can now write $\varepsilon \leq c\widetilde{\varepsilon}$ in a unified form, where

$$c = \begin{cases} 2\sqrt{2} \max\limits_{\nu_i \neq \nu} \dfrac{|\nu_i|}{|\nu_i + \tau - \rho|} & \text{for HJDSVD,} \\ 2\sqrt{2} \max\limits_{\nu_i \neq \nu} \dfrac{|\nu_i|}{|\nu_i + \tau - \rho'|} & \text{for RHJDSVD} \end{cases} \tag{5.3}$$

for $m > 1$, and $c = 1$ for $m = 1$ when solving (2.2) approximately.

For a not very small $\widetilde{\varepsilon}$, we may have $\varepsilon \geq 1$ in case c is large, which will make $(\widetilde{s}, \widetilde{t})$ have no accuracy as an approximation to the exact solution (s, t) to (2.2), so that $\widetilde{\mathcal{U}}_+$ and $\widetilde{\mathcal{V}}_+$ may have no improvement over $\mathcal{U}$ and $\mathcal{V}$. As a guard remedy, in order to make $\widetilde{\mathcal{U}}_+$ and $\widetilde{\mathcal{V}}_+$ have some improvements, we propose to use

$$\varepsilon \leq \min\{c\widetilde{\varepsilon}, 0.01\}. \tag{5.4}$$

However,

$$\varepsilon = \frac{\left\| \left[\begin{smallmatrix} \widetilde{s} \\ \widetilde{t} \end{smallmatrix}\right] - \left[\begin{smallmatrix} s \\ t \end{smallmatrix}\right] \right\|}{\left\| \left[\begin{smallmatrix} s \\ t \end{smallmatrix}\right] \right\|}$$

is an a prior error and uncomputable in practice, so that we cannot determine whether or not (5.4) is fulfilled for a prescribed $\widetilde{\varepsilon}$. Nevertheless, it is easy to justify that

$$\frac{\varepsilon}{\kappa(B')} \leq \|r_{in}\| \leq \kappa(B')\varepsilon, \tag{5.5}$$

where

$$\|r_{in}\| = \frac{1}{\|r\|} \left\| -r - \begin{bmatrix} I_M - P_u & \\ & I_N - P_v \end{bmatrix} \begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix} \begin{bmatrix} I_M - P_u & \\ & I_N - P_v \end{bmatrix} \begin{bmatrix} \widetilde{s} \\ \widetilde{t} \end{bmatrix} \right\| \tag{5.6}$$

is the computable relative residual norm of the approximate solution $[\widetilde{s}^T, \widetilde{t}^T]^T$ of (2.2), and $\kappa(B') = \|B'\| \|(B')^{-1}\|$ with

$$B' = B|_{(Q,Z)^{\perp\perp}} = \begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix}^{-1} \Big|_{(Q,Z)^{\perp\perp}}$$

being the restriction of B to the double orthogonal complement[3] of (Q, Z). Based on the two bounds in (5.5), we practically stop the inner iterations at each outer iteration when

$$\|r_{in}\| \leq \min\{c\widetilde{\varepsilon}, 0.01\} \tag{5.7}$$

[3] For arbitrary $Q \in \mathbb{R}^{M \times l}, Z \in \mathbb{R}^{N \times l}$, the double orthogonal complement of (Q, Z) is defined as $(Q, Z)^{\perp\perp} := \{(q, z) | q \in \mathbb{R}^M, z \in \mathbb{R}^N, q \perp Q, z \perp Z\}$.

Table 1

Properties of test matrices, where $nnz(A)$ is the number of nonzero entries in A, $\kappa(B')$ is defined by (5.8) for the chosen target τ when $k=0$, and $\sigma_{\max}$ and σ_2 are estimated by the MATLAB function svds. *The notation $+\infty$ indicates that A is rank deficient.*

Matrix	$M \times N$	$nnz(A)$	$\|A\|$	$\kappa(A)$	$\kappa(B')$
deter4	3235×9133	$19,231$	10.2	$3.71e+2$	13.7
lp_bnl2	2324×4486	$14,996$	$2.12e+2$	$7.77e+3$	$4.90e+2$
r05	5190×9690	$104,145$	18.2	$1.22e+2$	17.3
large	4282×8617	$20,635$	$4.04e+3$	$4.94e+5$	$2.25e+3$
gemat1	4929×10595	$46,591$	$2.34e+4$	$1.17e+8$	$9.97e+5$
tomographic1	73159×59498	$647,495$	6.98	$+\infty$	12.30
watson_1	201155×386992	$1,055,093$	20.59	$8.64e+2$	$7.58e+3$
degme	185501×659415	$8,127,528$	$2.24e+3$	$5.42e+2$	$1.52e+3$

for a given $\widetilde{\varepsilon}$. Equation (5.5) indicates that r_{in} is a reasonable replacement for ε when $\kappa(B')$ is fairly modest. We should remark that the lower and upper bounds in (5.5) are the estimates in the worst case for the relative residual norm r_{in} in terms of the a prior relative error ε.

Let us have a closer look at $\kappa(B')$, whose size has two effects: (i) it decides the convergence speed of the Krylov iterative solver MINRES [21], and the larger it is, the more slowly MINRES converges generally [3]; (ii) it decides how the a posterior relative residual norm r_{in} differs from the a prior error ε. The smaller $\kappa(B')$ is, the more reliable (5.7) is; conversely, for $\kappa(B')$ large, the a posterior replacement (5.7) may not be reliable. Notice that, when $(u,v)=(u^*,v^*)$ and $(U_c,V_c)=(U_k,V_k)$, we have

$$\kappa(B')=\frac{\sigma_{\max}(B')}{\sigma_{\min}(B')}=\frac{\max_{i=k+2,k+3,\dots,n}|\pm\sigma_i-\tau|}{\min_{i=k+2,k+3,\dots,n}|\pm\sigma_i-\tau|}=\frac{\sigma_{\max}+\tau}{|\sigma_{k+2}-\tau|}, \tag{5.8}$$

where $\sigma_{\max}=\max\{\sigma_{k+2},\sigma_{k+3},\dots,\sigma_n\}$. Therefore, the continuity tells us that $\kappa(B')\approx\frac{\sigma_{\max}+\tau}{|\sigma_{k+2}-\tau|}$ for $\sin\angle(u^*,u)=\mathcal{O}(\epsilon)$ and $\sin\angle(v^*,v)=\mathcal{O}(\epsilon)$. This result shows that the correction equation becomes (asymptotically) better conditioned as k increases.

6. Numerical experiments.

We report numerical experiments to confirm our theory. Table 1 lists the test matrices from [1] together with some of their basic properties, where $\kappa(B')$ is the right-hand side of (5.8) with $k=0$. For the matrices A with $M<N$, we apply the algorithms to their transposes A^T. We aim to show two points: (i) for fairly small $\widetilde{\varepsilon}=10^{-3}$ and 10^{-4}, the nonrestarted and restarted inexact HJDSVD and RHJDSVD algorithms behave (very much) like their exact counterparts; (ii) regarding the total inner iterations and overall efficiency, the inexact JDSVD type algorithms are substantially more efficient than their exact counterparts. We will compute the ℓ singular triplets for given τ's, where we take $\ell=1$ and $\ell=5$; we will report the experiments on two ℓ's, separately.

All the numerical experiments were performed on an Intel Core i7-7700 CPU 3.60GHz with the main memory 8GB, 4 cores, and 8 threads using the MATLAB R2017a with the machine precision $\epsilon_{\rm mach}=2.22\times10^{-16}$ under the Miscrosoft Windows 8 64-bit system. In our computations, 4 cores and 4 threads were actually used.

We denote by HJDSVD($\widetilde{\varepsilon}$) and RHJDSVD($\widetilde{\varepsilon}$) the inexact JDSVD algorithms for a given $\widetilde{\varepsilon}$. We use MINRES to solve the correction equation, (2.2) or (4.5), by taking the $(M+N)$-dimensional zero vector as an initial approximate solution. The code minres is from MATLAB R2017a. At each outer iteration of HJDSVD or RHJDSVD, we stop inner iterations when the stopping criterion (5.7) is fulfilled.

For outer iterations, we always take the initial vectors u_0 and v_0 to be the normalized M- and N-dimensional vectors whose elements are equal. We restart outer iterations after the maximum dimensions of $\mathcal{U}$ and $\mathcal{V}$ reach 20. The restarting technique used here is the thick-restart, i.e., when restarting, instead of one, we compute the best three approximate triplets $(\theta_i, u_i, v_i), i = 1, 2, 3$, in steps 5–7 of Algorithm 1. To do that, we compute three eigenvectors $f_i, i = 1, 2, 3$, of (F, G) associated with the largest three eigenvalues in magnitude, and each f_i corresponds to an approximate singular triplet $(\rho_i, \widetilde{u}_i, \widetilde{v}_i)$ for HJDSVD. As for RHJDSVD, we compute the eigenvector $\widehat{f}_i$ associated with the smallest eigenvalue of G' by taking $\rho = \rho_i,\ i = 1, 2, 3$ in (2.11), and each $\widehat{f}_i$ corresponds to an approximate singular triplet $(\rho'_i, \widehat{u}_i, \widehat{v}_i)$ for RHJDSVD. Then in the next restart cycle we use the computed three pairs of approximate left and right singular vectors to construct new initial $\mathcal{U}$ and $\mathcal{V}$ of dimension three, respectively, and expand them in the way described by Algorithm 1.

An approximate singular triplet $(\theta, u = Uc, v = Vd)$ obtained by the JDSVD type algorithms is claimed to have converged if the relative residual norm

$$\|r\| = \|r(\theta, u, v)\| \leq \|A\|_1 \cdot tol = \|A\|_1 \cdot 10^{-10}.$$

We stop outer iterations if all ℓ desired singular triplets have been found or the maximum outer iterations have been used.

In each of the exact algorithms, for the experimental purpose, we have also solved the correction equations and computed (3.3) by the LU factorization of $\left[\begin{smallmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{smallmatrix}\right]$, which is computed by the MATLAB built-in function lu, where α and β are obtained by the double orthogonality requirement of (s, t) and (u, v). However, due to excessive storage and/or computational cost, we must recall that it is generally unrealistic to perform the LU factorization when A is really large, as confirmed by our experiments. So in the exact JDSVD type algorithms it is practical to solve the correction equation, (2.2) or (4.5), by iterative solvers, in which we stop inner iterations when

$$\|r_{in}\| \leq 10^{-14}, \tag{6.1}$$

where $\|r_{in}\|$ is defined by (5.6), indicating as if (2.2) or (4.5) is solved exactly in finite precision arithmetic. In this case, we call the JDSVD algorithms iterative exact, denoted by "Iter. Exact."

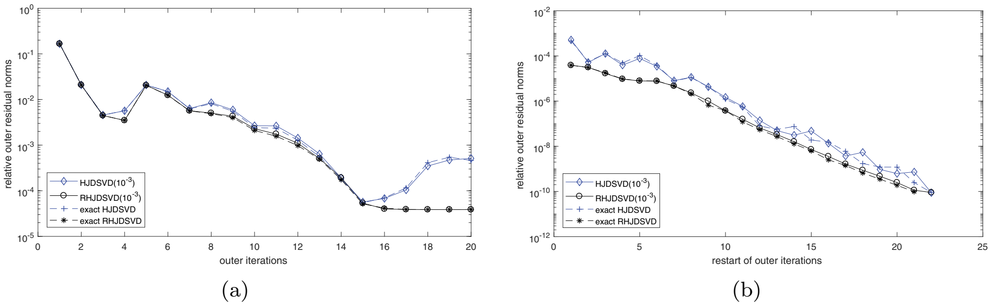
In all the tables, we denote by I_{out} and I_{in} the total numbers of outer and inner iterations, respectively, and by T_{cpu} the CPU time (in seconds) counted by the MATLAB recommended commands tic and toc. We point out that I_{in} is a reasonable measure of the overall efficiency of the JDSVD type algorithms, and it equals the total number of the matrices A and A^T-vector products used by MINRES. In contrast, T_{cpu} may be an unreliable measure of overall efficiency since it heavily depends on many factors, such as the computer used, the programming language used, the programming optimization, the computing environment, and the operating system.

We first test the matrix $A = $ deter4 with $\tau = 7$. The desired singular value $\sigma^* \approx 5.74$ is an interior singular value of A.

Table 2 lists the results obtained. Clearly, all the algorithms for $\widetilde{\varepsilon} = 10^{-3}, 10^{-4}$ are successful in computing σ and its corresponding singular vectors. Figure 1(a) and (b) depict the convergence curves of the inexact and exact JDSVDs during the first cycle and all cycles, respectively. We see that the restarted and nonrestarted HJDSVD(10^{-3}) and RHJDSVD(10^{-3}) behave quite like their exact counterparts, respectively.

Table 2
*deter*4 *with* $\tau = 7$.

Accuracy: $\widetilde{\varepsilon} = 10^{-3}$				Accuracy: Iter. Exact			
Algorithm	I_{out}	I_{in}	T_{cpu}	Algorithm	I_{out}	I_{in}	T_{cpu}
HJDSVD	430	7800	3.08	HJDSVD	427	36776	10.85
RHJDSVD	423	5673	2.47	RHJDSVD	418	34843	10.56
Accuracy: $\widetilde{\varepsilon} = 10^{-4}$				Accuracy: LU factorization			
Algorithm	I_{out}	I_{in}	T_{cpu}	Algorithm	I_{out}	I_{in}	T_{cpu}
HJDSVD	424	7722	2.91	HJDSVD	427	–	37.2
RHJDSVD	421	5728	2.34	RHJDSVD	418	–	36.6



(a) (b)

Fig. 1. *deter*4 *with* $\tau = 7$.

From Table 2, we observe that each inexact algorithm uses almost the same outer iterations as its exact counterpart does. This indicates that for $\widetilde{\varepsilon} = 10^{-3}, 10^{-4}$ the inexact JDSVD type algorithms mimic their exact versions very well. However, regarding the overall efficiency, compared with their iterative exact versions, we see that the inexact JDSVD algorithms cost only less than 22% of total inner iterations, or less than 28% of CPU time, to compute the desired singular triplet. A smaller $\widetilde{\varepsilon}$ is unnecessary since it cannot reduce outer iterations and in the meantime we have to solve the correction equations with higher accuracy, which will increase the total cost substantially. Finally, we mention in passing that each JDSVD with iterative exact is nearly three times faster than the corresponding LU based JDSVD, as indicated by T_{cpu}. But we must point out that it is hard to compare the computational efficiency of direct and iterative solvers for linear systems since iterative solvers are for large linear systems and their efficiency is highly problem dependent, while LU factorizations are generally suitable for small to medium sized problems.

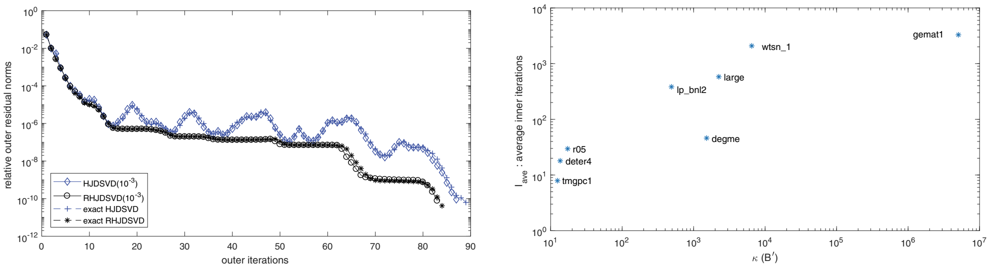
We can also see from Figure 1 and Table 2 that the restarted RHJDSVD algorithm uses fewer outer iterations than the restarted HJDSVD algorithm and improves the overall efficiency, as seen from I_{in} and T_{cpu}. Actually, we have observed from Figure 1(a) that (exact and inexact) RHJDSVD computed more accurate approximate singular triplets than HJDSVD at the 16th to the 20th outer iterations, and the latter diverged and delivered less accurate approximations, which confirms the better convergence of the refined harmonic extraction and the possible irregular convergence of the harmonic extraction.

We next test the matrix $A =$ lp_bnl2 with $\tau = 8.16$. The desired singular value $\sigma^* \approx 7.71$ is an interior one of A.

The results are displayed in Table 3 and Figure 2(a). We observe from Figure 2(a) that both nonrestarted and restarted HJDSVD(10^{-3}) and RHJDSVD(10^{-3}) behave almost the same as the corresponding exact HJDSVD and RHJDSVD, respectively,

TABLE 3
lp_bnl2 with $\tau = 8.16$.

Accuracy: $\widetilde{\varepsilon} = 10^{-3}$				Accuracy: Iter. Exact			
Algorithm	I_{out}	I_{in}	T_{cpu}	Algorithm	I_{out}	I_{in}	T_{cpu}
HJDSVD	87	30210	5.11	HJDSVD	89	122495	21.7
RHJDSVD	83	22620	4.16	RHJDSVD	84	113862	20.3
Accuracy: $\widetilde{\varepsilon} = 10^{-4}$				Accuracy: LU factorization			
Algorithm	I_{out}	I_{in}	T_{cpu}	Algorithm	I_{out}	I_{in}	T_{cpu}
HJDSVD	87	30044	5.41	HJDSVD	89	–	1.70
RHJDSVD	83	23209	4.24	RHJDSVD	84	–	1.53



(a) (b)

FIG. 2. (a) *lp_bnl2 with* $\tau = 8.16$; (b) I_{ave} *of HJDSVD*(10^{-3}) *versus* $\kappa(B')$ *in Table* 1.

and RHJDSVD converges more smoothly and gives a more accurate approximate singular triplet than HJDSVD. We also see from Table 3 that for $\widetilde{\varepsilon} = 10^{-3}$ and 10^{-4}, the inexact JDSVD algorithms use almost the same outer iterations to converge as their exact counterparts do. As far as the total inner iterations and overall efficiency are concerned, we can see from Table 3 that all our inexact JDSVD algorithms reduce more than 75% of total inner iterations and CPU time than their iterative exact counterparts. Apparently, a smaller $\widetilde{\varepsilon}$ is unnecessary. A final note is that RHJDSVD performs better than HJDSVD in terms of both outer and inner iterations.

For this problem, as the T_{cpu} indicate, each LU based JDSVD is several times faster than the corresponding JDSVD with iterative exact, which is opposite from the previous example.

We now test the other matrices. For the three middle-scale matrices A_1 = r05 with $\tau_1 = 4.75$, A_2 = large with $\tau_2 = 9.85$, and A_3 = gemat with $\tau_3 = 14.4$, the desired singular values $\sigma^* \approx 3.43$, $\sigma^* \approx 8.06$, and $\sigma^* \approx 14.38$ are all interior ones of the test matrices and are highly clustered with some other singular values; for the three large-scale matrices A_4 = tomographic1 (tmgpc1) with $\tau_4 = 8$, A_5 = watson_1 (wtsn_1) with $\tau_5 = 14$, and A_6 = degme with $\tau_6 = 5.56$, the desired $\sigma^* \approx 6.98$ is the largest one of A_4, $\sigma^* \approx 14.004$ is an interior one of A_5, and $\sigma^* \approx 4.13$ is the smallest one of A_6.

Since the matrices $A_i,\ i = 4, 5, 6$, are very large in our computer, it is unaffordable to implement the LU factorization of $\begin{bmatrix} -\tau I_M & A_i \\ A_i^T & -\tau I_N \end{bmatrix}$. So we only use MINRES to solve the correction equations involved in the exact JDSVD algorithms. For $A_i,\ i = 1, 2, 3$, we find that the outer iterations used by the exact JDSVD algorithms, where the correction equations are solved by the LU factorization of $\begin{bmatrix} -\tau I_M & A_i \\ A_i^T & -\tau I_N \end{bmatrix}$, are exactly the same as those by the iterative exact JDSVD algorithms, as they should be. Table 4 lists the details.

TABLE 4
Results on the other test matrices.

Matrix	Algorithm	$\widetilde{\varepsilon} = 10^{-3}$			$\widetilde{\varepsilon} = 10^{-4}$			Iter. Exact		
		I_{out}	I_{in}	T_{cpu}	I_{out}	I_{in}	T_{cpu}	I_{out}	I_{in}	T_{cpu}
r05	HJDSVD	89	2626	1.56	89	2716	1.61	89	14733	7.19
	RHJDSVD	82	2100	1.24	76	2010	1.17	80	12981	6.05
large	HJDSVD	147	85865	25.5	145	88449	25.5	154	492011	1.40e+2
	RHJDSVD	151	75227	21.7	146	75820	21.9	151	468380	1.36e+2
gemat1	HJDSVD	14	47801	20.2	13	50305	17.6	13	89327	35.9
	RHJDSVD	14	48205	19.7	14	54638	22.1	13	86521	35.8
tmgpc1	HJDSVD	22	176	1.70	22	185	1.63	22	1085	7.55
	RHJDSVD	22	174	1.92	22	183	1.69	22	1082	7.25
wstn_1	HJDSVD	13	27181	8.65e+2	13	30033	9.68e+2	13	44306	1.94e+3
	RHJDSVD	13	26989	8.73e+2	13	30017	9.65e+2	13	44462	1.94e+3
degme	HJDSVD	11	509	37.3	10	568	41.3	10	2395	1.67e+2
	RHJDSVD	11	510	36.2	10	564	40.6	10	2387	2.23e+2

For these six matrices, we have observed very similar phenomena to the previous examples, so we make comments on them together. We find that all the inexact JDSVD type algorithms behave almost the same as their iterative exact counterparts and use almost the same or very comparable outer iterations as the latter ones do. It is seen from Table 4 that when computing the desired interior singular triplets of A_3 and A_5, our inexact JDSVD algorithms reduce more than 32% of total inner iterations and 38% of CPU time, compared with their iterative exact counterparts. The reductions of total inner iterations are up to 76% for A_6, and 81% for A_1, A_2, and A_4, and the corresponding reductions of total CPU time are up to 95% and 73%, respectively. These are substantial savings, compared with their iterative exact counterparts.

Clearly, a fairly small $\widetilde{\varepsilon} \in [10^{-4}, 10^{-3}]$ is enough for the inexact JDSVD algorithms to mimic their exact counterparts and reduce the computational costs substantially, and a smaller $\widetilde{\varepsilon}$ is unnecessary.

Summarizing the previous experiments, we conclude that HJDSVD and RHJDSVD are suitable for computing both an interior and an extreme singular triplet.

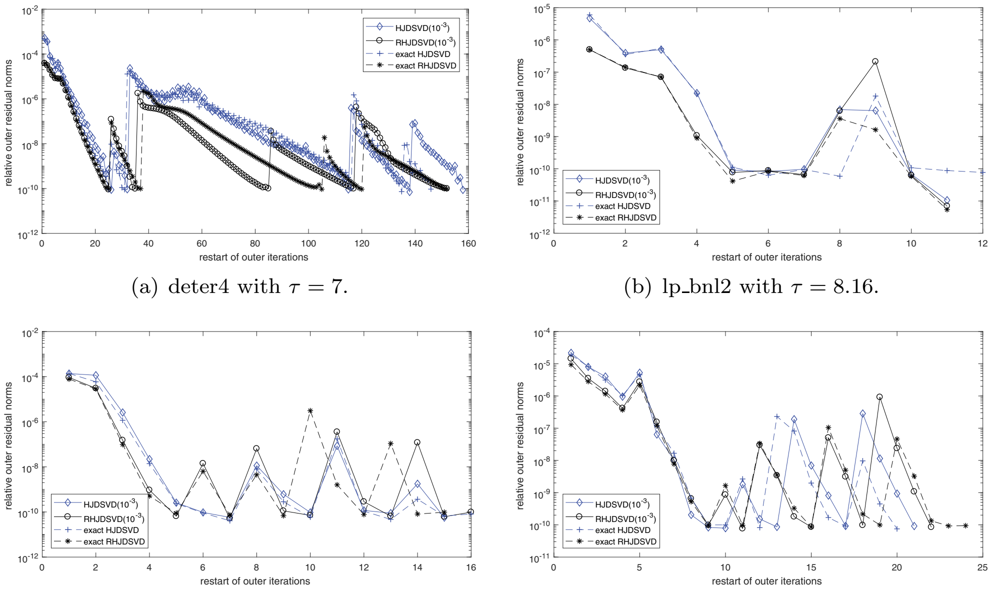
In the following we compute the five singular values of the matrices in Table 1 nearest to the given target τ's given in the previous experiments and the corresponding left and right singular vectors.

Table 5 gives the results on the eight test matrices. Figure 3 depicts the convergence curves of the inexact and exact JDSVDs of all cycles for computing the five singular triplets of deter4 with $\tau = 7$, lp_bnl2 with $\tau = 8.16$, r05 with $\tau = 4.75$, and large with $\tau = 9.85$. Notice that each algorithm computes the desired singular triplets successively and it computes the next one after the previous one has converged. As a result, its convergence curve has five stages and contains five convergence points (valleys), and each stage computes one singular triplet.

We see that the restarted HJDSVD(10^{-3}) and RHJDSVD(10^{-3}) behave quite like their exact counterparts. For all the test matrices, we see from Table 5 that our inexact JDSVD type algorithms use very comparable outer iterations to their iterative exact counterparts. Furthermore, for the matrices gemat1 and wstn_1, our inexact JDSVD algorithms reduce more than 34% and 22% of total inner iterations, respectively, compared with their exact counterparts; the reductions of total inner reductions for deter4, lp_bnl2, r05, large, tmgpc1, and degme are more than 75%, substantial savings, compared with the exact JDSVD algorithms. Clearly, our inexact JDSVD algorithms with deflation can mimic their exact counterparts well with a

TABLE 5
Results on the computation of the five singular triplets of the test matrices in Table 1.

Matrix	Algorithm	$\widetilde{\varepsilon}=10^{-3}$			$\widetilde{\varepsilon}=10^{-4}$			Iter. Exact		
		I_{out}	I_{in}	T_{cpu}	I_{out}	I_{in}	T_{cpu}	I_{out}	I_{in}	T_{cpu}
deter4	HJDSVD	2610	47045	20.1	1897	34374	14.3	2324	190378	63.2
	RHJDSVD	2508	29669	14.4	2377	28289	13.7	2380	192268	64.8
lp_bnl2	HJDSVD	114	39063	7.02	122	41427	7.75	134	180803	33.8
	RHJDSVD	122	34391	6.64	114	32221	6.09	114	152770	28.4
r05	HJDSVD	186	5515	3.36	192	5724	3.41	184	30264	15.2
	RHJDSVD	192	4969	3.28	176	4629	2.92	177	28521	14.0
large	HJDSVD	289	170847	54.5	285	172054	53.6	273	874840	2.65e+2
	RHJDSVD	311	160411	50.2	300	156601	48.7	316	988726	3.05e+2
gemat1	HJDSVD	27	82899	37.0	25	78417	33.7	23	149002	63.8
	RHJDSVD	31	94866	41.5	27	88256	37.6	23	145875	64.3
tmgpc1	HJDSVD	80	589	6.53	78	582	6.58	75	3525	28.1
	RHJDSVD	77	566	6.81	75	561	6.16	75	3531	27.5
wstn_1	HJDSVD	25	42834	1.47e+3	22	41995	1.44e+3	18	59750	2.76e+3
	RHJDSVD	25	44491	1.54e+3	22	41933	1.58e+3	17	56947	2.50e+3
degme	HJDSVD	57	2445	2.03e+2	58	2749	2.28e+2	55	13323	1.12e+3
	RHJDSVD	56	2173	1.80e+2	56	2425	2.01e+2	52	12377	1.28e+3



(a) deter4 with $\tau = 7$.

(b) lp_bnl2 with $\tau = 8.16$.

(c) r05 with $\tau = 4.75$.

(d) large with $\tau = 9.85$.

FIG. 3. *Computing the five singular triplets of the matrices with a given target* τ.

fairly small $\widetilde{\varepsilon} \in [10^{-4}, 10^{-3}]$ and meanwhile reduce the computational cost substantially.

In addition, from Figure 3 we have observed faster and smoother convergence of the RHJDSVD algorithms than the HJDSVD algorithms when computing each of the desired singular triplets. We can also see from Table 5 that the total outer iterations used by RHJDSVD($\widetilde{\varepsilon}$) and HJDSVD($\widetilde{\varepsilon}$) are very comparable, but for most of the matrices, i.e., deter4, lp_bnl2, r05, large, tmgpc1, and degme, RHJDSVD($\widetilde{\varepsilon}$) uses fewer total inner iterations and less CPU time than HJDSVD($\widetilde{\varepsilon}$). Therefore, as a whole, we conclude that, when computing more than one singular triplet, RHJDSVD is generally more robust and efficient than HJDSVD.

We now get insight into the role that $\kappa(B')$ plays in the inexact JDSVD algorithms. We take HJDSVD(10^{-3}) for computing one singular triplet as an example. Denote by $I_{ave} = I_{in}/I_{out}$ the average inner iterations per outer iteration. Note that a smaller I_{ave} indicates a faster, on average, convergence of MINRES. To make it clearer to see how $\kappa(B')$ influences the convergence speed of inner iterations, we mark the matrix names in the plot of $\kappa(B')$ versus I_{ave} in Figure 2(b). We observe a trend from the figure and Table 1 that the larger $\kappa(B')$ is, the more inner iterations per outer iteration are needed to achieve the convergence, i.e., the more slowly MINRES converges. We have observed similar phenomena for the exact and inexact HJDSVD and RHJDSVD, which confirms our analysis at the end of section 5.

Finally, let us make further comments on the correction equations (2.2) and (4.5). This system is typically symmetric indefinite, and it may be ill conditioned when the desired singular value σ is an interior one, so that MINRES or other Krylov iterative solvers may converge slowly. Therefore, preconditioning is naturally appealing. However, it is hard to obtain an effective preconditioner for MINRES when (2.2) or (4.5) is highly indefinite and ill conditioned. Using the MATLAB built-in function ilu, we have tried the sparse incomplete LU factorizations of $\begin{bmatrix} -\tau I_M & A \\ A^T & -\tau I_N \end{bmatrix}$ with *setup.droptol* $= 0.1$ and 0.01. With such preconditioners, the preconditioned correction equations are nonsymmetric, and we use the Krylov solver BiCGStab algorithm [21] to solve them. Unfortunately, we have found that such preconditioners are ineffective. For most of the test problems, the preconditioned BiCGStab is not even competitive with the unpreconditioned MINRES and uses more inner iterations. So we do not report the results on the preconditioned BiCGStab.

7. Conclusions.

We have proposed harmonic and refined harmonic JDSVD methods for computing several singular triplets of a large matrix A. By a rigorous one-step analysis, we have proved for the first time that, provided the correction equations (2.2) and (4.5) involved in the JDSVD methods are only solved with low or modest accuracy, the inexact JDSVD methods mimic their exact counterparts well. Based on the theory, we have proposed general-purpose practical stopping criteria for inner iterations involved in the two inexact JDSVD type methods. We should point out that the theory and criteria also work for the standard and refined inexact JDSVD methods, but these methods are inferior to HJDSVD and RHJDSVD for computing interior singular triplets. Moreover, our results apply to the methods in [5, 6].

Numerical experiments have confirmed our theory. We have tested a number of problems and compared the nonrestarted and restarted inexact JDSVD algorithms with their exact counterparts. We have found that the inexact JDSVD algorithms indeed mimic the exact JDSVD algorithms very well when the correction equations are solved with the modest or low accuracy $10^{-4} \sim 10^{-3}$. A great advantage of the inexact JDSVD type algorithms is that they reduce the computational cost very substantially, compared with their iterative exact versions. Furthermore, the experiments have illustrated that RHJDSVD generally outperforms HJDSVD.

Acknowledgment. We would like to thank two referees for valuable comments and suggestions that improved the presentation of the paper substantially.

REFERENCES

[1] T. A. Davis and Y. Hu, *The University of Florida sparse matrix collection*, ACM Trans. Math. Software, 38 (2011), pp. 1–25, also available online from http://www.cise.ufl.edu/research/sparse/matrices/.

[2] G. H. Golub and C. F. van Loan, *Matrix Computations*, 4th ed., The John Hopkins University Press, Baltimore, MD, 2012.

[3] A. Greenbaum, *Iterative Methods for Solving Linear Systems*, Front. Appl. Math. 17, SIAM, Philadephia, 1997.

[4] V. Hernandes, J. E. Roman, and A. Tomas, *A robust and efficient parallel SVD solver based on restarted Lanczos bidiagonalization*, Electron. Trans. Numer. Anal., 31 (2008), pp. 68–85.

[5] M. E. Hochstenbach, *A Jacobi–Davidson type SVD method*, SIAM J. Sci. Comput., 23 (2001), pp. 606–628.

[6] M. E. Hochstenbach, *Harmonic and refined extraction methods for the singular value problem, with applications in least squares problems*, BIT, 44 (2004), pp. 721–754.

[7] M. E. Hochstenbach and Y. Notay, *Controlling inner iteration in the Jacobi–Davidson method*, SIAM J. Matrix Anal. Appl., 31 (2009), pp. 460–477.

[8] Z. Jia, *The refined harmonic Arnoldi method and an implicitly restarted refined algorithm for computing interior eigenpairs of large matrices*, Appl. Numer. Math., 42 (2002), pp. 489–512.

[9] Z. Jia, *Some theoretical comparisons of refined Ritz vectors and Ritz vectors*, Sci. China Ser. A, 47 (2004), pp. 222–233.

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

[11] Z. Jia, *Using cross–product matrices to compute the SVD*, Numer. Algorithms, 42 (2006), pp. 31–61.

[12] Z. Jia and C. Li, *Inner iterations in the shift-invert residual Arnoldi method and the Jacobi–Davidson method*, Sci. China Math., 57 (2014), pp. 1733–1752.

[13] Z. Jia and C. Li, *Harmonic and refined harmonic shift-invert residual Arnoldi and Jacobi–Davidson methods for interior eigenvalue problems*, J. Comput. Appl. Math., 282 (2015), pp. 83–97.

[14] Z. Jia and D. Niu, *An implicitly restarted refined bidiagonalization Lanczos method for computing a partial singular value decomposition*, SIAM J. Matrix Anal. Appl., 25 (2003), pp. 246–265.

[15] Z. Jia and D. Niu, *A refined harmonic Lanczos bidiagonalization method and an implicitly restarted algorithm for computing the smallest singular triplets of large matrices*, SIAM J. Sci. Comput., 32 (2010), pp. 714–744.

[16] Z. Jia and G. W. Stewart, *An analysis of the Rayleigh–Ritz method for approximating eigenspaces*, Math. Comp., 70 (2001), pp. 637–647.

[17] E. Kokiopoulou, C. Bekas, and E. Gallopoulos, *Computing smallest singular triplets with implicitly restarted Lanczos bidiagonalization*, Appl. Numer. Math., 49 (2004), pp. 39–61.

[18] D. Niu and X. Yuan, *A harmonic Lanczos bidiagonalization method for computing interior singular triplets of large matrices*, Appl. Math. Comput., 218 (2012), pp. 7459–7467.

[19] Y. Notay, *Combination of Jacobi–Davidson and conjugate gradients for the partial symmetric eigenproblem*, Numer. Linear Algebra Appl., 9 (2002), pp. 21–44.

[20] B. N. Parlett, *The Symmetric Eigenvalue Problem*, Classics Appl. Math. 20, SIAM, Philadelphia, 1998.

[21] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed., SIAM, Philadelphia, 2003.

[22] Y. Saad, *Numerical Methods for Large Eigenvalue Problems*, SIAM, Philadelphia, 2011.

[23] H. D. Simon and H. Zha, *Low-rank matrix approximation using the Lanczos bidiagonalization process with applications*, SIAM J. Sci. Comput., 21 (2000), pp. 2257–2274.

[24] G. L. G. Sleijpen and H. A. van der Vorst, *A Jacobi–Davidson iteration method for linear eigenvalue problems*, SIAM Rev., 42 (2000), pp. 267–293.

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

[26] M. Stoll, *A Krylov–Schur approach to the truncated SVD*, Linear Algebra Appl., 436 (2012), pp. 2795–2806.

[27] G. Wu, *The convergence of harmonic Ritz vectors and harmonic Ritz values, revisited*, SIAM J. Matrix Anal. Appl., 38 (2017), pp. 118–133.

[28] L. Wu, R. Romero, and A. Stathopoulos, *PRIMME_SVDS: A high-performance preconditioned SVD solver for accurate large-scale computations*, SIAM J. Sci. Comput., 39 (2017), pp. S248–S271.

[29] L. Wu and A. Stathopoulos, *A preconditioned hybrid SVD method for accurately computing singular triplets of large matrices*, SIAM J. Sci. Comput., 37 (2015), pp. S365–S388.