An Inexact Newton-Type Method for Inverse Singular Value Problems

Zheng-Jian Bai* Shufang Xu†

Revised Version: March 8, 2008

Abstract

In this paper, an inexact Newton-type approach is proposed for solving inverse singular value problems. We show that the method converges superlinearly. This method can reduce significantly the oversolving problem of the Newton-type method and improve the efficiency. Numerical experiments is also presented to illustrate our results.

Keywords. Inverse problem, singular value, Newton's method, root-convergence rate
AMS subject classifications. 15A29, 15A18, 65F15, 65F18

1 Introduction

Let $\{A_i\}_{i=0}^n$ be $n+1$ real m-by-n matrices, $m \geq n$. For any vector $\mathbf{c} = (c_1, c_2, \ldots, c_n)^T \in \mathbb{R}^n$, we define

$$A(\mathbf{c}) := A_0 + \sum_{i=1}^{n} c_i A_i, \tag{1}$$

and denote the singular values of $A(\mathbf{c})$ by $\{\sigma_i(\mathbf{c})\}_{i=1}^n$, where $\sigma_1(\mathbf{c}) \geq \sigma_2(\mathbf{c}) \geq \cdots \geq \sigma_n(\mathbf{c}) \geq 0$. The inverse singular value problem (ISVP) is defined as follows: Given n nonnegative real numbers $\sigma_1^* \geq \sigma_2^* \geq \cdots \geq \sigma_n^*$, find $\mathbf{c}^* = (c_1^*, c_2^*, \ldots, c_n^*)^T \in \mathbb{R}^n$ such that $\sigma_i(\mathbf{c}^*) = \sigma_i^*$ for $i = 1, 2, \ldots, n$.

The special ISVP was originally proposed by Chu [8]. In general, An ISVP concerns the construction of a structured matrix from its singular values. A general ISVP can be seen as an inverse problem, which arises in different applications such as the determination of mass distributions, orbital mechanics, irrigation theory, computed tomography, circuit theory, ect. [16, 20, 25]. For symmetric matrices, the ISVPs are essentially the same as the IEPs, which play an important role in many applications, see for instance [9, 10] and [11, 27]. Recently, there some different ISVPs have been considered such as the low rank update of singular values [7] and the ISVP in some quadratic group [22]. However, considering the rectangular matrices (i.e., the case when $m \geq n$), the ISVP can naturally seen as the extension of the IEPs, which is a complicated but interesting topic for further study.

*Department of Information and Computational Mathematics, Xiamen University, Xiamen 361005, People's Republic of China (zjbai@xmu.edu.cn). This author's research was partially supported by the National Natural Science Foundation of China Grant 10601043 and Program for New Century Excellent Talents in Xiamen University.

†School of Mathematical Sciences, Peking University, Beijing 100871, People's Republic of China (xsf@pku.edu.cn). This author's research was partially supported by the NSFC under grant 10571007.

Even though the solvability issue for an ISVP is very complicated, the effective numerical algorithms for solving the problem can still be developed. The second method proposed in [8] is designed for the ISVP, which is essentially a Newton-type method applied the nonlinear system

$$\mathbf{f}(\mathbf{c}) = \boldsymbol{\sigma}^*, \tag{2}$$

where $\boldsymbol{\sigma}^* = (\sigma_1^*, \ldots, \sigma_n^*)^T$, $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$ is a nonlinear operator defined by

$$\mathbf{f}(\mathbf{c}) := \big(\sigma_1(\mathbf{c}), \ldots, \sigma_n(\mathbf{c})\big)^T, \quad \mathbf{c} = (c_1, c_2, \ldots, c_n)^T \in \mathbb{R}^n. \tag{3}$$

In each Newton (outer) iteration of a Newton-type method, we need to solve an approximate Jacobian equation. If the dimension n is large, directly solving such a linear system will be very expensive. We can reduce the cost by using iterative methods (the inner iterations). Although an iterative method can reduce the complexity, it may *oversolve* the approximate Jacobian equation in the sense that the last tens or hundreds inner iterations before convergence may not improve the convergence of the outer Newton iterations [13]. The inexact Newton-type method aims to stop the inner iteration with the relative residual less than a given tolerance. By choosing suitable stopping criteria, we can reduce the total cost of the whole inner-outer iterations. In fact, the approximate Jacobian equation may be solved inexactly in order that the Newton method converges fast.

In this paper, we propose an inexact version of the Newton-type method for solving the ISVP. This approach is motivated by two recent papers due to Chan, Chung, and Xu [6] and Bai, Chan, and Morini [3] (see also [2]). In [6] and [3], the inexact versions of Methods II and III in [15] are provided for solving IEPs where the easily computable tolerances are derived for the Jacobian equation and the superlinear convergence is preserved. In our inexact method for the ISVP, we provide a new stopping tolerance which is available handily. We show that our inexact method converges superlinearly. Also, our method can minimize the oversolving problem of the Newton-type method in [8] as showed in the later numerical tests.

Finally, we point out that, in the practical implementation, we should combine the global methods (e.g. the homotopy method [1], [11, pp. 46–47], [27, pp.256–62], and the the continuous method in [8, Section 2]), which converges globally but slowly, with our inexact iterative method. In these global methods, our inexact method is employed as the corrector step where a good start point is provided by the global strategies.

This paper is outlined as follows. In Section 2 we recall the Newton-type method for solving the ISVP. In Section 3 we introduce our inexact version. In Section 4 we give the convergence analysis of our method. In Section 5 we report some numerical results. Finally, we give some concluding remarks in Section 6.

Throughout this paper, we will use some notations and definitions as follows. Let $\|\cdot\|$ and $\|\cdot\|_F$ denote the Euclidean vector norm or its corresponding induced matrix norm and the matrix Frobenius norm, respectively. I is the identity matrix of an appropriate dimension. Define $\boldsymbol{\sigma}(\mathbf{c}) = (\sigma_1(\mathbf{c}), \ldots, \sigma_n(\mathbf{c}))^T$ and $\Sigma_* = \mathrm{diag}(\sigma_1^*, \ldots, \sigma_n^*) \in \mathbb{R}^{m\times n}$. Denote by $\mathscr{O}(n)$ the set of all orthogonal matrices in $\mathbb{R}^{n\times n}$. Let $U(\mathbf{c}) := [\mathbf{u}_1(\mathbf{c}), \ldots, \mathbf{u}_m(\mathbf{c})] \in \mathscr{O}(m)$ and $V(\mathbf{c}) := [\mathbf{v}_1(\mathbf{c}), \ldots, \mathbf{v}_n(\mathbf{c})] \in \mathscr{O}(n)$ be two orthogonal matrices of the left singular vectors and the right singular vectors of $A(\mathbf{c})$, respectively. Finally, suppose $\mathbf{c}^*$ is a solution of the ISVP.

2 The Newton-type Method

In this section, we briefly recall the Newton-type method proposed by Chu [8]. For simplicity, we assume that all the singular values $\{\sigma_i^*\}_{i=1}^n$ are positive and distinct, i.e., $\sigma_1^* > \sigma_2^* \cdots > \sigma_n^* > 0$. Define the affine subspace $\mathscr{A} \equiv \{A(\mathbf{c})|\mathbf{c} \in \mathbb{R}^n\}$ and the surface $\mathscr{G}_s(\Sigma_*) \equiv \{P\Sigma_* Q^T | P \in \mathscr{O}(m), Q \in \mathscr{O}(n)\}$, i.e. the set of all matrices in $\mathbb{R}^{m\times n}$ with singular values $\{\sigma_i^*\}_{i=1}^n$. Then, solving the ISVP is equivalent to finding an intersection of $\mathscr{G}_s(\Sigma_*)$ and $\mathscr{A}$.

The Newton iterative approach in [8] is simply a generalization of Method III in [15]. Let $\mathbf{c}^*$ be a solution of the ISVP. Suppose that $Y_k \in \mathscr{G}_s(\Sigma_*)$, there exist $P_k \in \mathscr{O}(n)$ and $Q_k \in \mathscr{O}(n)$ such that

$$Y_k = P_k \Sigma_* Q_k^T. \tag{4}$$

The new iterate $\mathbf{c}^{k+1} \in \mathbb{R}^n$ is determined by seeking a $\mathscr{A}$-intercept $A(\mathbf{c}^{k+1})$ from a line that is tangent to the manifold $\mathscr{G}_s(\Sigma_*)$ at Y_k. To get the intercept, it is required to find two skew-symmetric matrices $C_k \in \mathbb{R}^{m\times m}$, $D_k \in \mathbb{R}^{n\times n}$ and a vector $\mathbf{c}^{k+1} \in \mathbb{R}^n$ such that

$$Y_k + C_k Y_k - Y_k D_k = A(\mathbf{c}^{k+1}).$$

By (4), we have

$$\Sigma_* + \widetilde{C}_k \Sigma_* - \Sigma_* \widetilde{D}_k = P_k^T A(\mathbf{c}^{k+1}) Q_k \equiv X_k, \tag{5}$$

where $\widetilde{C}_k = P_k^T C_k P_k$ and $\widetilde{D}_k = Q_k^T D_k Q_k$ are two skew-symmetric matrices.

We observe from (5) that the lower-right corner of size $(m-n)$-by-$(m-n)$ in $\widetilde{C}_k$ can be arbitrary. In [8], these entries are set to be identically zeros, i.e.,

$$[\widetilde{C}_k]_{ij} = 0 \quad \text{for } n+1 \le i \ne j \le m. \tag{6}$$

In fact, different allocations of these free entries have a direct effect on the convergence speed of the Newton-type method, see [4].

For $1 \le i = j \le n$, by (5), we obtain the following equation

$$J_k \mathbf{c}^{k+1} = \boldsymbol{\sigma}^* - \mathbf{a}^k, \tag{7}$$

where

$$\begin{aligned} [J_k]_{ij} &\equiv (\mathbf{p}_i^k)^T A_j \mathbf{q}_i^k, \quad 1 \le i, j \le n, && (8)\\ \boldsymbol{\sigma}^* &\equiv (\sigma_1^*, \ldots, \sigma_n^*)^T, &&\\ a_i^k &\equiv (\mathbf{p}_i^k)^T A_0 \mathbf{q}_i^k, \quad 1 \le i \le n, && (9) \end{aligned}$$

where $\mathbf{p}_i^k$ and $\mathbf{q}_i^k$ are the column vectors of P_k and Q_k, respectively. If the matrix J_k is nonsingular, then we can solve (7) for the vector $\mathbf{c}^{k+1}$.

Next, the skew-symmetric matrices $\widetilde{C}_k$ and $\widetilde{D}_k$ are obtained by comparing the "off-diagonal" entries in (5). For $n+1 \le i \le m, 1 \le j \le n$, we have

$$[\widetilde{C}_k]_{ij} = -[\widetilde{C}_k]_{ji} = \frac{[X_k]_{ij}}{\sigma_j^*}. \tag{10}$$

For $1 \le i < j \le n$, we get

$$\begin{aligned} [X_k]_{ij} &= \sigma_j^* [\widetilde{C}_k]_{ij} - \sigma_i^* [\widetilde{D}_k]_{ij}, \\ [X_k]_{ji} &= \sigma_i^* [\widetilde{C}_k]_{ji} - \sigma_j^* [\widetilde{D}_k]_{ji} = -\sigma_i^* [\widetilde{C}_k]_{ij} + \sigma_j^* [\widetilde{D}_k]_{ij}, \end{aligned}$$

which lead to

$$[\widetilde{C}_k]_{ij} = -[\widetilde{C}_k]_{ji} = \frac{\sigma_i^*[X_k]_{ji} + \sigma_j^*[X_k]_{ij}}{(\sigma_j^*)^2 - (\sigma_i^*)^2}, \tag{11}$$

$$[\widetilde{D}_k]_{ij} = -[\widetilde{D}_k]_{ji} = \frac{\sigma_i^*[X_k]_{ij} + \sigma_j^*[X_k]_{ji}}{(\sigma_j^*)^2 - (\sigma_i^*)^2}. \tag{12}$$

This completes the intercept step.

Finally, it remains to lift the intercept $A(\mathbf{c}^{k+1})$ back to $\mathscr{G}_s(\Sigma_*)$. To do so, define the lift

$$Y_{k+1} \equiv P_{k+1}\Sigma_* Q_{k+1}^T,$$

where the orthogonal matrices P_{k+1} and Q_{k+1} are defined by

$$P_{k+1} = P_k S_k \quad \text{and} \quad Q_{k+1} = Q_k T_k.$$

Here, S_k and T_k are the Cayley transforms

$$S_k \equiv \big(I + \frac{1}{2}\widetilde{C}_k\big)\big(I - \frac{1}{2}\widetilde{C}_k\big)^{-1} \quad \text{and} \quad T_k \equiv \big(I + \frac{1}{2}\widetilde{D}_k\big)\big(I - \frac{1}{2}\widetilde{D}_k\big)^{-1}.$$

Overall we have:

Algorithm I: (The Newton-type Method)

1. *Given $\mathbf{c}^0$, compute the singular values $\{\sigma_i(\mathbf{c}^0)\}_{i=1}^n$, the normalized left singular vectors $\{\mathbf{p}_i(\mathbf{c}^0)\}_{i=1}^m$, and the normalized right singular vectors $\{\mathbf{q}_i(\mathbf{c}^0)\}_{i=1}^n$ of $A(\mathbf{c}^0)$. Let*

$$\begin{aligned} P_0 &= [\mathbf{p}_1^0, \ldots, \mathbf{p}_m^0] = [\mathbf{p}_1(\mathbf{c}^0), \ldots, \mathbf{p}_m(\mathbf{c}^0)] \in \mathscr{O}(m), \\ Q_0 &= [\mathbf{q}_1^0, \ldots, \mathbf{q}_n^0] = [\mathbf{q}_1(\mathbf{c}^0), \ldots, \mathbf{q}_n(\mathbf{c}^0)] \in \mathscr{O}(n), \\ \boldsymbol{\sigma}^0 &:= \boldsymbol{\sigma}(A(\mathbf{c}^0)) = (\sigma_1(\mathbf{c}^0), \ldots, \sigma_n(\mathbf{c}^0))^T. \end{aligned}$$

2. *For $k = 0, 1, 2, \ldots,$ until convergence, do:*

   (a) *Form the approximate Jacobian matrix J_k by (8) and $\mathbf{a}^k$ by (9).*

   (b) *Solve $\mathbf{c}^{k+1}$ from the approximate Jacobian equation (7).*

   (c) *Form the matrix $A(\mathbf{c}^{k+1})$ by (1).*

   (d) *Form the matrix $X_k \equiv P_k^T A(\mathbf{c}^{k+1}) Q_k$.*

   (e) *Compute the skew-symmetric matrices $\widetilde{C}_k$ and $\widetilde{D}_k$ by (6) and (10)–(12).*

   (f) *Compute $P_{k+1} = [\mathbf{p}_1^{k+1}, \ldots, \mathbf{p}_m^{k+1}]$ and $Q_{k+1} = [\mathbf{q}_1^{k+1}, \ldots, \mathbf{q}_n^{k+1}]$ by solving*

$$\big(I + \frac{1}{2}\widetilde{C}_k\big)P_{k+1}^T = \big(I - \frac{1}{2}\widetilde{C}_k\big)P_k^T, \tag{13}$$

$$\big(I + \frac{1}{2}\widetilde{D}_k\big)Q_{k+1}^T = \big(I - \frac{1}{2}\widetilde{D}_k\big)Q_k^T. \tag{14}$$

This approach is showed to converge at least quadratically in the root sense in [4]. For the definition of root-convergence rate, see Section 3 or [21, Chapter 9]. We point out that in each outer iteration (i.e. Step 2), we have to solve the linear equations (7) and (13)–(14). If the dimension of the problem is large, one may reduce the computational cost by solving these equations iteratively. One may certainly expect to solve equations (13)–(14) iteratively with only a few iterations. This is because that both $\|\widetilde{C}_k\|$ and $\|\widetilde{D}_k\|$ converge to zeros, see [4, equation (44)]. However, iterative methods may oversolve the approximate Jacobian equation (7) in the sense that, at each outer (Newton) step, the last tens or hundreds inner iterations may not contribute the convergence of outer iterations. To reduce the redundant inner iterations sharply is our goal in next section.

3 The Inexact Newton-type Method

In this section, we propose an inexact version of Algorithm I. To decrease the computational complexity, we use iterative methods to solve equations (7) and (13)–(14). Especially, we solve equation (7) inexactly. That is, we will find an explicit stopping tolerance for (7), and then investigate the convergence of the resulted procedure.

For a general nonlinear equation $\mathbf{h}(\mathbf{x}) = \mathbf{0}$, the stopping criterion of inexact Newton methods is usually given in terms of $\mathbf{h}(\mathbf{x})$, see for instance [13, 14]. By (2), this will involve computing the exact singular values $\{\sigma_i(\mathbf{c}^k)\}_{i=1}^n$ of $A(\mathbf{c}^k)$ which are costly to compute. In this paper, we will replace $\{\sigma_i(\mathbf{c}^k)\}_{i=1}^n$ by the readily computational quantities as defined in (17) and (20) below. We will show in Section 4 that this replacement will retain the superlinear convergence.

Algorithm II: The Inexact Newton-type Method

1. *Given $\mathbf{c}^0$, compute the singular values $\{\sigma_i(\mathbf{c}^0)\}_{i=1}^n$, the orthogonal left singular vectors $\{\mathbf{u}_i(\mathbf{c}^0)\}_{i=1}^m$ and right singular vectors $\{\mathbf{v}_i(\mathbf{c}^0)\}_{i=1}^n$ of $A(\mathbf{c}^0)$. Let*

$$\begin{aligned} U_0 &= [\mathbf{u}_1^0, \ldots, \mathbf{u}_m^0] = [\mathbf{u}_1(\mathbf{c}^0), \ldots, \mathbf{u}_m(\mathbf{c}^0)], \\ V_0 &= [\mathbf{v}_1^0, \ldots, \mathbf{v}_n^0] = [\mathbf{v}_1(\mathbf{c}^0), \ldots, \mathbf{v}_n(\mathbf{c}^0)], \\ \boldsymbol{\sigma}^0 &= (\sigma_1(\mathbf{c}^0), \sigma_2(\mathbf{c}^0), \ldots, \sigma_n(\mathbf{c}^0))^T. \end{aligned}$$

2. *For $k = 0, 1, 2, \ldots,$ until convergence, do*

   (a) *Form the approximate Jacobian matrix J_k and the vector $\mathbf{a}^k$:*

$$[J_k]_{ij} = (\mathbf{u}_i^k)^T A_j \mathbf{v}_i^k, \quad 1 \le i, j \le n. \tag{15}$$

$$a_i^k = (\mathbf{u}_i^k)^T A_0 \mathbf{v}_i^k, \quad 1 \le i \le n.$$

   (b) *Solve c^{k+1} inexactly from the approximate Jacobian equation:*

$$J_k \mathbf{c}^{k+1} = \boldsymbol{\sigma}^* - \mathbf{a}^k + \mathbf{r}^k, \tag{16}$$

   until the residual $\mathbf{r}^k$ satisfies

$$\|\mathbf{r}^k\| \le \frac{\|\boldsymbol{\sigma}^k - \boldsymbol{\sigma}^*\|^\beta}{\|\boldsymbol{\sigma}^*\|^\beta}, \quad \beta \in (1, 2]. \tag{17}$$

   (c) *Form the matrix $A(\mathbf{c}^{k+1})$ given by (1).*

   (d) *Form the matrix $W_k \equiv U_k^T A(\mathbf{c}^{k+1}) V_k$.*

(e) *Compute the skew-symmetric matrices H_k and K_k by*

$$
\begin{aligned}
&[H_k]_{ij} = 0 \quad \text{for } n+1 \le i \ne j \le m,\\
&[H_k]_{ij} = -[H_k]_{ji} = \frac{[W_k]_{ij}}{\sigma_j^*}, \quad \text{for } n+1 \le i \le m, 1 \le j \le n,\\
&[H_k]_{ij} = -[H_k]_{ji} = \frac{\sigma_i^*[W_k]_{ji} + \sigma_j^*[W_k]_{ij}}{(\sigma_j^*)^2 - (\sigma_i^*)^2}, \quad \text{for } 1 \le i < j \le n,\\
&[K_k]_{ij} = -[K_k]_{ji} = \frac{\sigma_i^*[W_k]_{ij} + \sigma_j^*[W_k]_{ji}}{(\sigma_j^*)^2 - (\sigma_i^*)^2}, \quad \text{for } 1 \le i < j \le n.
\end{aligned}
$$

(f) *Compute $U_{k+1} = [\mathbf{u}_1^{k+1}, \ldots, \mathbf{u}_m^{k+1}]$ and $V_{k+1} = [\mathbf{v}_1^{k+1}, \ldots, \mathbf{v}_n^{k+1}]$ by solving*

$$\big(I + \frac{1}{2}H_k\big)U_{k+1}^T = \big(I - \frac{1}{2}H_k\big)U_k^T, \tag{18}$$

$$\big(I + \frac{1}{2}K_k\big)V_{k+1}^T = \big(I - \frac{1}{2}K_k\big)V_k^T. \tag{19}$$

(h) *Compute $\boldsymbol{\sigma}^{k+1} = (\sigma_1^{k+1}, \ldots, \sigma_n^{k+1})^T$ by*

$$\sigma_i^{k+1} = (\mathbf{u}_i^{k+1})^T A(\mathbf{c}^{k+1})\mathbf{v}_i^{k+1}, \quad \text{for } 1 \le i \le n. \tag{20}$$

On Algorithm II, we give some remarks as follows.

Remark 3.1 *Since U_0 and V_0 are both orthogonal, and H_k and K_k are all skew-symmetric, the matrices U_k and V_k generated by the Cayley transforms in (18)–(19) should be orthogonal, i.e.,*

$$U_k^T U_k = I \quad \text{and} \quad V_k^T V_k = I, \quad k = 0, 1, \ldots.$$

To guarantee the orthogonality of U_k and V_k, equations (18)–(19) cannot be solved inexactly. However, we will see in Section 4 that both $\|H_k\|$ and $\|K_k\|$ converge to zeros as the initial guess $\mathbf{c}^0$ is close to $\mathbf{c}^*$ sufficiently (see (74)). Then the matrices on the left-hand sides of (18) and (19) approach to the identity matrices in the limits. Thus, we can expect to solve (18) and (19) precisely by iterative methods with just a few iterations.

Remark 3.2 *In Algorithm II, the solution of (16) will be the costly step. In the next section, we will establish that the convergence rate of Algorithm II is equal to β given in (17).*

4 Convergence Analysis

In this section, we will present the convergence results for Algorithm II. In what follows, we suppose that the given singular values $\{\sigma_i^*\}_{i=1}^n$ are all positive and distinct. Let the singular value decomposition of $A(\mathbf{c}^*)$ be given by $A(\mathbf{c}^*) = U(\mathbf{c}^*)\Sigma_* V(\mathbf{c}^*)^T$ with $U(\mathbf{c}^*) \in \mathscr{O}(m)$ and $V(\mathbf{c}^*) \in \mathscr{O}(n)$. Let $\mathbf{c}^* = (c_1^*, \ldots, c_n^*)^T \in \mathbb{R}^n$ be a solution of the ISVP with the given data $\{A_i\}_{i=0}^n$ and $\{\sigma_i^*\}_{i=1}^n$. By [27, Theorem 1.9.3], there exists a neighborhood of $\mathbf{c}^*$ where the singular values $\{\sigma_i(\mathbf{c})\}_{i=1}^n$ are all distinct and analytic. In this neighborhood, the function $\mathbf{f}(\mathbf{c})$ defined by (3) is analytic. Differentiating the relations

$$\sigma_i(\mathbf{c}) = \mathbf{u}_i(\mathbf{c})^T A(\mathbf{c})\mathbf{v}_i(\mathbf{c}), \quad \mathbf{u}_i(\mathbf{c})^T\mathbf{u}_i(\mathbf{c}) = 1, \quad \mathbf{v}_i(\mathbf{c})^T\mathbf{v}_i(\mathbf{c}) = 1$$

give rise to

$$\frac{\partial \sigma_i(\mathbf{c})}{\partial c_j} = \mathbf{u}_i(\mathbf{c})^T A_j \mathbf{v}_i(\mathbf{c}). \tag{21}$$

Thus the Jacobian of $\mathbf{f}$ is given by

$$J(\mathbf{c}) = [\mathbf{u}_i(\mathbf{c})^T A_j \mathbf{v}_i(\mathbf{c})]. \tag{22}$$

In this paper, we assume that the ISVP satisfies the conditions below:

i) There exists a solution $\mathbf{c}^*$ such that $A(\mathbf{c}^*)$ has the given singular values $\{\sigma_i^*\}_{i=1}^n$.

ii) The Jacobian $J(\mathbf{c}^*)$ defined in (22) is nonsingular.

To show our results, let $\mathbf{c}^k$ be the kth iterate produced by Algorithm II. Partition Σ_*, $U(\mathbf{c}^*)$ and U_k as

$$\Sigma_* = \begin{bmatrix} \Sigma_{*1} \\ 0 \end{bmatrix}, \quad U(\mathbf{c}^*) = [U_{11}(\mathbf{c}^*), U_{12}(\mathbf{c}^*)], \quad \text{and} \quad U_k = [U_{11}^{(k)}, U_{12}^{(k)}],$$

where $\Sigma_{*1} \in \mathbb{R}^{n\times n}$, $U_{11}(\mathbf{c}^*) \in \mathbb{R}^{m\times n}$, and $U_{11}^{(k)} \in \mathbb{R}^{m\times n}$. At the k-th stage, we define

$$E_1^{(k)} := U_{11}^{(k)} - U_{11}(\mathbf{c}^*) \quad \text{and} \quad E_2^{(k)} := V_k - V(\mathbf{c}^*).$$

4.1 Preliminary Lemmas

In this subsection, we prove some preliminary results which are necessary for the convergence analysis of our inexact method. We first give four lemmas that are shown in the literature.

Lemma 4.1 [17, Corollary 8.6.2] *If B and $B+E$ are in $\mathbb{R}^{m\times n}$ with $m \ge n$, then, for any $1 \le k \le n$,*

$$|\sigma_k(B+E) - \sigma_k(B)| \le \|E\|,$$

where $\sigma_k(B)$ denotes the kth largest singular value of B.

Lemma 4.2 [4, Lemma 2] *For any $\mathbf{c}, \bar{\mathbf{c}} \in \mathbb{R}^n$, we have*

$$\|A(\mathbf{c}) - A(\bar{\mathbf{c}})\| \le \alpha \|\mathbf{c} - \bar{\mathbf{c}}\|,$$

where $A(\mathbf{c})$ is defined in (1) and $\alpha = (\sum\limits_{i=1}^{n} \|A_i\|^2)^{1/2}$.

Lemma 4.3 [4, Lemma 5] *If $E \in \mathbb{R}^{n\times n}$ and $\|E\| < 1$, then $I - \frac{1}{2}E$ is nonsingular and*

$$\|(I + \frac{1}{2}E)(I - \frac{1}{2}E)^{-1} - (I+E)\| \le \|E\|^2.$$

Lemma 4.4 [4, Lemma 6] *Let $B \in \mathbb{R}^{n\times n}$ and $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_n) \in \mathbb{R}^{n\times n}$ with $\sigma_1 > \sigma_2 > \cdots > \sigma_n > 0$. If the n-by-n skew-symmetric matrices H and K satisfy that*

$$H\Sigma - \Sigma K = B,$$

then we have

$$\|H\| \le \frac{2n\sigma_1}{d}\|B\| \quad \text{and} \quad \|K\| \le \frac{2n\sigma_1}{d}\|B\|,$$

where $d = \min_{i\ne j} |\sigma_i^2 - \sigma_j^2|$.

We now show the uniform invertibility of the approximate Jacobian J_k defined by (15).

Lemma 4.5 *Let J_k and $J(\mathbf{c}^*)$ be defined as in (15) and (22), respectively. Then $\|J_k - J(\mathbf{c}^*)\| = O(\|E_1^{(k)}\| + \|E_2^{(k)}\|)$. Hence if $J(\mathbf{c}^*)$ is nonsingular, then there exist positive numbers ξ and c such that if $\max\{\|E_1^{(k)}\|, \|E_2^{(k)}\|\} \le \xi$, then J_k is nonsingular and*

$$\|J_k^{-1}\| \le c.$$

Proof: The first part follows easily from the formula of J_k and $J(\mathbf{c}^*)$, and the second part follows from the continuity of matrix inverses, cf. [6] or [27, p. 249, Equation (4.6.11)]. □

Remark 4.6 *We observe from Lemma 4.5 that the assumption that the Jacobian matrix $J(\mathbf{c}^*)$ is nonsingular can be replaced by the assumption that the approximate Jacobian matrices J_k defined in Algorithm II are uniformly invertible, i.e, $\limsup_{k\to\infty}\{\|J_k^{-1}\|\} < \infty$, see the similar assumption for IEPs [15, Assumption 3.1 (ii)].*

We point out that the main difference between Algorithm II and Algorithm I is that we solve (16) approximately rather than exactly as in (7). By comparing with (5), we see that the skew-symmetric matrices H_k and K_k and the vector $\mathbf{c}^{k+1}$ of Algorithm II should be defined by

$$\Sigma_* + H_k\Sigma_* - \Sigma_* K_k = U_k^T A(\mathbf{c}^{k+1})V_k - R_k, \tag{23}$$

where $R_k = \mathrm{diag}(r_1^k, \ldots, r_n^k) \in \mathbb{R}^{m\times n}$ with the vector $\mathbf{r}^k$ given in (17).

In the following lemma, we estimate $\|\mathbf{c}^{k+1} - \mathbf{c}^k\|$, $\|\boldsymbol{\sigma}^k - \boldsymbol{\sigma}^*\|$, $\|H_k\|$, $\|K_k\|$, $\|U_{k+1} - U_k\|$, and $\|V_{k+1} - V_k\|$ in terms of $\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|$, $\|H_{k-1}\|$, and $\|K_{k-1}\|$ when the later quantities are sufficiently small. We mainly employ the equations (23) and (18)–(19) as our argument. In particular, by (18)–(19), the new estimates of the matrices of the left singular vectors and the right singular vectors are given by

$$U_k = U_{k-1}(I + \frac{1}{2}H_{k-1})(I - \frac{1}{2}H_{k-1})^{-1} \quad \text{and} \quad V_k = V_{k-1}(I + \frac{1}{2}K_{k-1})(I - \frac{1}{2}K_{k-1})^{-1}.$$

Now, multiplying the left-hand side and the right-hand-side of (23) by the factors $(I + \frac{1}{2}H_{k-1})(I - \frac{1}{2}H_{k-1})^{-1}$ and $(I + \frac{1}{2}K_{k-1})(I - \frac{1}{2}K_{k-1})^{-1}$, respectively, leads to

$$U_k^T A(\mathbf{c}^k)V_k = \Sigma_* + R_{k-1} + \Omega_{k-1},$$

where $\Omega_{k-1} = O(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2)$ as in (32) below. By combining this equation with (23), we obtain the following result.

Lemma 4.7 *Let the given singular values $\{\sigma_i^*\}_{i=1}^n$ be positive and distinct. Suppose also the matrices J_k defined in (15) are uniformly invertible, i.e, $\limsup_{k\to\infty}\{\|J_k^{-1}\|\} < \infty$. Then there exist three positive numbers ϵ_1, ϵ_2, and ϵ_3 such that the conditions $\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\| \le \epsilon_1$, $\|H_{k-1}\| \le \epsilon_2$ and $\|K_{k-1}\| \le \epsilon_3$ imply*

$$\|\mathbf{c}^{k+1} - \mathbf{c}^k\| = O(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2), \tag{24}$$

$$\sqrt{\|\boldsymbol{\sigma}^k - \boldsymbol{\sigma}^*\|^2 + \|H_k\|^2 + \|K_k\|^2} = O(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2), \tag{25}$$

$$\|U_{k+1} - U_k\| = O(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2), \tag{26}$$

$$\|V_{k+1} - V_k\| = O(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2). \tag{27}$$

Proof: Let

$$\Phi_k = (I + \frac{1}{2}H_k)(I - \frac{1}{2}H_k)^{-1} \quad \text{and} \quad \Psi_k = (I + \frac{1}{2}K_k)(I - \frac{1}{2}K_k)^{-1}.$$

Then by (18)–(19), we have

$$U_k = U_{k-1}\Phi_{k-1} \quad \text{and} \quad V_k = V_{k-1}\Psi_{k-1}. \tag{28}$$

By Lemma 4.3, if $\|H_{k-1}\| \le 1$ and $\|V_{k-1}\| \le 1$, then we can write

$$\Phi_{k-1} = I + H_{k-1} + F_{k-1} \quad \text{and} \quad \Psi_{k-1} = I + K_{k-1} + G_{k-1}, \tag{29}$$

where

$$\|F_{k-1}\| \le \|H_{k-1}\|^2 \quad \text{and} \quad \|G_{k-1}\| \le \|K_{k-1}\|^2. \tag{30}$$

Notice from (23) that

$$U_{k-1}^T A(\mathbf{c}^k)V_{k-1} = \Sigma_* + H_{k-1}\Sigma_* - \Sigma_* K_{k-1} + R_{k-1}.$$

Then by (28)–(29), a simple calculation gives rise to

$$U_k^T A(\mathbf{c}^k)V_k = \Sigma_* + R_{k-1} + \Omega_{k-1}, \tag{31}$$

where

$$\begin{aligned}\Omega_{k-1} \;\; = \;\; & F_{k-1}^T\left(\Sigma_* + H_{k-1}\Sigma_* - \Sigma_* K_{k-1} + R_{k-1}\right)\left(I + K_{k-1} + G_{k-1}\right) \\ & +(I - H_{k-1})\left(\Sigma_* + H_{k-1}\Sigma_* - \Sigma_* K_{k-1} + R_{k-1}\right)G_{k-1} \\ & +\left(H_{k-1}\Sigma_* - \Sigma_* K_{k-1} + R_{k-1}\right)K_{k-1} - H_{k-1}\left(H_{k-1}\Sigma_* - \Sigma_* K_{k-1} + R_{k-1}\right) \\ & -H_{k-1}\left(\Sigma_* + H_{k-1}\Sigma_* - \Sigma_* K_{k-1} + R_{k-1}\right)K_{k-1}.\end{aligned}$$

Thus if $\|H_{k-1}\|$, $\|V_{k-1}\|$, and $\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|$ are small enough, then by (30) and (17), we get

$$\|\Omega_{k-1}\| = O(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^{2\beta} + \|H_{k-1}\|^2 + \|K_{k-1}\|^2). \tag{32}$$

Equating the diagonal elements of (31) leads to

$$\boldsymbol{\sigma}^k = \boldsymbol{\sigma}^* + \mathbf{r}^{k-1} + O(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^{2\beta} + \|H_{k-1}\|^2 + \|K_{k-1}\|^2),$$

and by (17), we obtain

$$\|\boldsymbol{\sigma}^k - \boldsymbol{\sigma}^*\| = O(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^{\beta} + \|H_{k-1}\|^2 + \|K_{k-1}\|^2), \tag{33}$$

On the other hand, taking the difference between (23) and (31) yields

$$U_k^T\big(A(\mathbf{c}^{k+1}) - A(\mathbf{c}^k)\big)V_k = H_k\Sigma_* - \Sigma_* K_k + R_k - R_{k-1} - \Omega_{k-1}. \tag{34}$$

Now, based on (34), we will verify (24)–(27). To show (24), we note that the diagonal equations of (34) give rise to

$$J_k(\mathbf{c}^{k+1} - \mathbf{c}^k) = \mathbf{r}^k - \mathbf{r}^{k-1} + O(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^{2\beta} + \|H_{k-1}\|^2 + \|K_{k-1}\|^2).$$

By the uniform invertibility of J_k, we have

$$\|\mathbf{c}^{k+1} - \mathbf{c}^k\| = O(\|\mathbf{r}^k\| + \|\mathbf{r}^{k-1}\| + O(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^{2\beta} + \|H_{k-1}\|^2 + \|K_{k-1}\|^2). \tag{35}$$

By (17), (33), and (35), we obtain (24).

To get (25), let

$$Z \equiv U_k^T\big(A(\mathbf{c}^{k+1}) - A(\mathbf{c}^k)\big)V_k - R_k + R_{k-1} + \Omega_{k-1} = \left[\begin{array}{c} Z_{11} \\ Z_{21} \end{array}\right]_{m-n}^{n}.$$

Noting that H_k has the form

$$H_k = \left[\begin{array}{cc} H_{11}^{(k)} & -H_{21}^{(k)T} \\ H_{21}^{(k)} & 0 \end{array}\right], \quad H_{11}^{(k)} \in \mathbb{R}^{n\times n}.$$

Then, by (34), we obtain

$$H_{11}^{(k)}\Sigma_{*1} - \Sigma_{*1}K_k = Z_{11}, \tag{36}$$

$$H_{21}^{(k)}\Sigma_{*1} = Z_{21}. \tag{37}$$

By Lemma 4.4, it follows from (36) that

$$\|H_{11}^{(k)}\| = O(\|Z_{11}\|) = O(\|Z\|), \tag{38}$$

$$\|K_k\| = O(\|Z_{11}\|) = O(\|Z\|). \tag{39}$$

Next, by (37), we have

$$\|H_{21}^{(k)}\| = O(\|Z_{21}\|) = O(\|Z\|).$$

This, together with (38), yields

$$\|H_k\| = O(\|H_{11}^{(k)}\| + 2\|H_{21}^{(k)}\|) = O(\|Z\|), \tag{40}$$

By Lemma 4.2, it follows from (17), (24), and (32)–(33) that

$$\begin{aligned} \|Z\| &= O(\|\mathbf{c}^{k+1} - \mathbf{c}^k\| + \|\mathbf{r}^k\| + \|\mathbf{r}^{k-1}\| + \|\Omega_{k-1}\|) \\ &= O(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2). \end{aligned} \tag{41}$$

By (33) and (39)–(41), we get (25).

Now we prove (26)–(27). By lemma 4.3, it follows from (28) and (25) that

$$\|U_{k+1} - U_k\| = \|U_k(\Phi_k - I)\| = O(\|H_k\|) = O(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2),$$

$$\|V_{k+1} - V_k\| = \|V_k(\Psi_k - I)\| = O(\|K_k\|) = O(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2).$$

□

Finally, we estimate the errors in $\{\mathbf{u}_i(\mathbf{c}^k)\}_{i=1}^n$ and $\{\mathbf{v}_i(\mathbf{c}^k)\}_{i=1}^n$ in terms of $\|\mathbf{c}^k - \mathbf{c}^*\|$.

Lemma 4.8 [4, Lemma 4] *Let the given singular values $\{\sigma_i^*\}_{i=1}^n$ be positive and distinct. Let the vectors $\mathbf{u}_i(\mathbf{c}^k)$ and $\mathbf{v}_i(\mathbf{c}^k)$ stand for the unit left and unit right singular vectors of $A(\mathbf{c}^k)$ respectively. Then there exist positive numbers ϵ_4 and γ such that, if $\|\mathbf{c}^k - \mathbf{c}^*\| \le \epsilon_4$, we have*

$$\begin{aligned} \|[\mathbf{u}_1(\mathbf{c}^k), \ldots, \mathbf{u}_n(\mathbf{c}^k)] - U_{11}(\mathbf{c}^*)\| &\le \gamma\|\mathbf{c}^k - \mathbf{c}^*\|, \\ \|[\mathbf{v}_1(\mathbf{c}^k), \ldots, \mathbf{v}_n(\mathbf{c}^k)] - V(\mathbf{c}^*)\| &\le \gamma\|\mathbf{c}^k - \mathbf{c}^*\|. \end{aligned}$$

4.2 R-Convergence Rate of Algorithm II

In the following, we show that the root-convergence rate of our method is at least β. Here, we recall the definition of root-convergence, see [21, Chap. 9].

Definition 4.9 *Let $\{\mathbf{x}^k\}$ be a sequence with limit $\mathbf{x}^*$. Then the numbers*

$$R_p\{\mathbf{x}^k\} = \begin{cases} \limsup_{k\to\infty} \|\mathbf{x}^k - \mathbf{x}^*\|^{1/k}, & \text{if } p = 1, \\ \limsup_{k\to\infty} \|\mathbf{x}^k - \mathbf{x}^*\|^{1/p^k}, & \text{if } p > 1, \end{cases} \tag{42}$$

are the root-convergence factors of $\{\mathbf{x}^k\}$. The quantity

$$O_R(\mathbf{x}^*) = \begin{cases} \infty, & \text{if } R_p\{\mathbf{x}^k\} = 0, \forall p \in [1,\infty), \\ \inf\{p \in [1,\infty) | R_p\{\mathbf{x}^k\} = 1\}, & \text{otherwise}, \end{cases} \tag{43}$$

is called the root-convergence rate of $\{\mathbf{x}^k\}$.

First we prove that our method is locally convergent. Based on the results in Lemma 4.7, we use the mathematical induction and the similar strategies used in the proof of Lemma 4.7 to estimate the quantities $\|\mathbf{c}^{k+1} - \mathbf{c}^k\|$, $\|\boldsymbol{\sigma}^k - \boldsymbol{\sigma}^*\|$, $\|H_k\|$, $\|K_k\|$, $\|U_{k+1} - U_k\|$, and $\|V_{k+1} - V_k\|$ in terms of $\|\mathbf{c}^0 - \mathbf{c}^*\|$ when the initial guess $\mathbf{c}^0$ is sufficiently close to the solution $\mathbf{c}^*$.

Theorem 4.10 *Let the given singular values $\{\sigma_i^*\}_{i=1}^n$ be positive and distinct and the Jacobian matrix $J(\mathbf{c}^*)$ defined by (22) be nonsingular. Then there exists $\epsilon > 0$ such that if $\|\mathbf{c}^0 - \mathbf{c}^*\| \leq \epsilon$, the sequence $\{\mathbf{c}^k\}$ generated by Algorithm II converges.*

Proof: Suppose that the matrices J_k defined in (15) are uniformly invertible, $\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\| \leq \epsilon_5$, $\|H_{k-1}\| \leq \epsilon_5$, and $\|K_{k-1}\| \leq \epsilon_5$, where $\epsilon_5 \equiv \min\{1, \epsilon_1, \epsilon_2, \epsilon_3\}$ with ϵ_1, ϵ_2, and ϵ_3 given in Lemma 4.7. By Lemma 4.7, there exists a constant $\tau > 1$ such that for any $k \geq 1$,

$$\|\mathbf{c}^{k+1} - \mathbf{c}^k\| \leq \tau(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2), \tag{44}$$

$$\sqrt{\|\boldsymbol{\sigma}^k - \boldsymbol{\sigma}^*\|^2 + \|H_k\|^2 + \|K_k\|^2} \leq \tau(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2), \tag{45}$$

$$\|U_{k+1} - U_k\| \leq \tau(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2), \tag{46}$$

$$\|V_{k+1} - V_k\| \leq \tau(\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2). \tag{47}$$

We note by Lemma 4.5 that if $\max\{\|E_1^{(k)}\|, \|E_2^{(k)}\|\} \leq \xi$, then the matrix J_k defined in (15) are uniformly invertible.

Let

$$\phi = \max\left\{1,\, 3\tau,\, \sqrt{n}\,\alpha,\, \big(1 + \frac{1}{\|\boldsymbol{\sigma}^*\|^\beta}\big)\sqrt{n}\,\alpha(c\alpha + 1)\big(\frac{2n\sigma_1^*}{d_*} + \frac{2}{\sigma_n^*}\big)\right\} > 1,$$

where $d_* = \min_{i\neq j} |(\sigma_i^*)^2 - (\sigma_j^*)^2|$. By (44)–(47), it is derived that

$$\begin{aligned} &\max\{\|\mathbf{c}^{k+1} - \mathbf{c}^k\|, \|\boldsymbol{\sigma}^k - \boldsymbol{\sigma}^*\|, \|H_k\|, \|K_k\|, \|U_{k+1} - U_k\|, \|V_{k+1} - V_k\|\} \\ &\leq \phi \max\{\|\boldsymbol{\sigma}^{k-1} - \boldsymbol{\sigma}^*\|^\beta, \|H_{k-1}\|^\beta, \|K_{k-1}\|^\beta\}, \quad k = 1, 2, \ldots. \end{aligned} \tag{48}$$

Next, we will use the mathematical induction to prove that if $\|\mathbf{c}^0 - \mathbf{c}^*\| \leq \epsilon$, where

$$\epsilon \equiv \min\left\{1, \frac{1}{\sqrt{n}\,\alpha}, \epsilon_4, \frac{\xi}{2\gamma}, \frac{\epsilon_5}{\phi}, \frac{\xi}{8\phi}, \left(\frac{\xi}{4+\xi}\right)^{\beta/\ln\beta}, \frac{1}{\phi^{\beta^2/(\beta-1)^2}}\right\} < 1, \tag{49}$$

then for each $k \geq 1$, the following inequalities hold:

$$\max\{\|E_1^{(k)}\|, \|E_2^{(k)}\|\} \leq \xi, \tag{50}$$

$$\begin{aligned}&\max\{\|\mathbf{c}^{k+1} - \mathbf{c}^k\|, \|\boldsymbol{\sigma}^k - \boldsymbol{\sigma}^*\|, \|H_k\|, \|K_k\|, \|U_{k+1} - U_k\|, \|V_{k+1} - V_k\|\} \\ &\leq \phi^{1+\beta+\cdots+\beta^k} \|\mathbf{c}^0 - \mathbf{c}^*\|^{\beta^k},\end{aligned} \tag{51}$$

$$\max\{\|\mathbf{c}^{k+1} - \mathbf{c}^k\|, \|\boldsymbol{\sigma}^k - \boldsymbol{\sigma}^*\|, \|H_k\|, \|K_k\|, \|U_{k+1} - U_k\|, \|V_{k+1} - V_k\|\} \leq \epsilon. \tag{52}$$

We first estimate $\|\boldsymbol{\sigma}^0 - \boldsymbol{\sigma}^*\|$, $\|H_0\|$, and $\|K_0\|$ in terms of $\|\mathbf{c}^0 - \mathbf{c}^*\|$. By Lemmas 4.1 and 4.2, we have

$$\max_i |\sigma_i(\mathbf{c}^0) - \sigma_i^*| \leq \|A(\mathbf{c}^0) - A(\mathbf{c}^*)\| \leq \alpha \|\mathbf{c}^0 - \mathbf{c}^*\|.$$

Thus

$$\|\boldsymbol{\sigma}^0 - \boldsymbol{\sigma}^*\| \leq \sqrt{n} \max_i |\sigma_i(\mathbf{c}^0) - \sigma_i^*| \leq \sqrt{n}\alpha \|\mathbf{c}^0 - \mathbf{c}^*\| \leq \phi \|\mathbf{c}^0 - \mathbf{c}^*\| \leq \epsilon_5. \tag{53}$$

Let $\Sigma_0 \equiv \mathrm{diag}(\sigma_0(\mathbf{c}^0), \ldots, \sigma_n(\mathbf{c}^0)) \in \mathbb{R}^{m\times n}$. It is easily to know that

$$U_0^T A(\mathbf{c}^0) V_0 = \Sigma_0. \tag{54}$$

Notice from (23) that

$$U_0^T A(\mathbf{c}^1) V_0 = \Sigma_* + H_0 \Sigma_* - \Sigma_* K_0 + R_0. \tag{55}$$

Taking the difference between (54) and (55) yields

$$U_0^T \left(A(\mathbf{c}^1) - A(\mathbf{c}^0)\right) V_0 = \Sigma_* - \Sigma_0 + H_0 \Sigma_* - \Sigma_* K_0 + R_0. \tag{56}$$

The diagonal equations of (56) give rise to

$$J_0(\mathbf{c}^1 - \mathbf{c}^0) = \boldsymbol{\sigma}^* - \boldsymbol{\sigma}^0 + \mathbf{r}^0. \tag{57}$$

By Lemma 4.8 and using (49), we have

$$\max\{\|E_1^{(0)}\|, \|E_2^{(0)}\|\} = \max\{\|U_{11}^{(0)} - U_{11}(\mathbf{c}^*)\|, \|V_0 - V(\mathbf{c}^*)\|\} \leq \gamma \|\mathbf{c}^0 - \mathbf{c}^*\| \leq \frac{\xi}{2}. \tag{58}$$

Thus J_0 is nonsingular and $\|J_0^{-1}\| \leq c$. Therefore, by (17), (53), and (57), we have

$$\left\|\mathbf{c}^1 - \mathbf{c}^0\right\| \leq c(\|\boldsymbol{\sigma}^* - \boldsymbol{\sigma}^0\| + \|\mathbf{r}^0\|) \leq \left(1 + \frac{1}{\|\boldsymbol{\sigma}^*\|^\beta}\right) \sqrt{n}\, c\, \alpha \|\mathbf{c}^0 - \mathbf{c}^*\|. \tag{59}$$

To estimate $\|H_0\|$ and $\|K_0\|$ in terms of $\|\mathbf{c}^0 - \mathbf{c}^*\|$, let

$$Z \equiv U_0^T \left(A(\mathbf{c}^1) - A(\mathbf{c}^0)\right) V_0 - \Sigma_* + \Sigma_0 - R_0 = \left[\begin{array}{c} Z_{11} \\ Z_{21} \end{array}\right]_{m-n}^{n}.$$

Then, by Lemma 4.2 and using (17), (53), and (59), we have

$$\begin{aligned}\|Z\| &\leq \|A(\mathbf{c}^1) - A(\mathbf{c}^0)\| + \|\boldsymbol{\sigma}^* - \boldsymbol{\sigma}^0\| + \|\mathbf{r}^0\| \leq \alpha \|\mathbf{c}^1 - \mathbf{c}^0\| + \left(1 + \frac{1}{\|\boldsymbol{\sigma}^*\|^\beta}\right) \|\boldsymbol{\sigma}^* - \boldsymbol{\sigma}^0\| \\ &\leq \left(1 + \frac{1}{\|\boldsymbol{\sigma}^*\|^\beta}\right) \sqrt{n}\alpha(c\alpha + 1) \|\mathbf{c}^0 - \mathbf{c}^*\|.\end{aligned} \tag{60}$$

Notice that H_0 can be partitioned into the form

$$H_0 = \begin{bmatrix} H_{11}^{(0)} & -H_{21}^{(0)T} \\ H_{21}^{(0)} & 0 \end{bmatrix}, \quad H_{11}^{(0)} \in \mathbb{R}^{n\times n}.$$

Then from (56), we obtain

$$H_{11}^{(0)}\Sigma_{*1} - \Sigma_{*1}K_0 = Z_{11}, \tag{61}$$

and

$$H_{21}^{(0)}\Sigma_{*1} = Z_{21}, \tag{62}$$

By Lemma 4.4, it follows from (61) that

$$\|H_{11}^{(0)}\| \leq \frac{2n\sigma_1^*}{d_*}\|Z_{11}\|, \tag{63}$$

$$\|K_0\| \leq \frac{2n\sigma_1^*}{d_*}\|Z_{11}\| \leq \frac{2n\sigma_1^*}{d_*}\|Z\|. \tag{64}$$

On the other hand, by (62), we have

$$\|H_{21}^{(0)}\| \leq \frac{1}{\sigma_n^*}\|Z_{21}\|.$$

This, together with (63), yields

$$\|H_0\| \leq \|H_{11}^{(0)}\| + 2\|H_{21}^{(0)}\| \leq \frac{2n\sigma_1^*}{d_*}\|Z_{11}\| + \frac{2}{\sigma_n^*}\|Z_{21}\| \leq \Big(\frac{2n\sigma_1^*}{d_*} + \frac{2}{\sigma_n^*}\Big)\|Z\|. \tag{65}$$

By (60) and (64)–(65), we get

$$\|H_0\| \leq \Big(1 + \frac{1}{\|\boldsymbol{\sigma}^*\|^\beta}\Big)\sqrt{n}\alpha(c\alpha+1)\Big(\frac{2n\sigma_1^*}{d_*} + \frac{2}{\sigma_n^*}\Big)\|\mathbf{c}^0 - \mathbf{c}^*\| \leq \phi\|\mathbf{c}^0 - \mathbf{c}^*\| \leq \epsilon_5, \tag{66}$$

$$\|K_0\| \leq \Big(1 + \frac{1}{\|\boldsymbol{\sigma}^*\|^\beta}\Big)\sqrt{n}\alpha(c\alpha+1)\frac{2n\sigma_1^*}{d_*}\|\mathbf{c}^0 - \mathbf{c}^*\| \leq \phi\|\mathbf{c}^0 - \mathbf{c}^*\| \leq \epsilon_5. \tag{67}$$

By Lemma 4.3, we have from (28) that

$$\|U_1 - U_0\| \leq \|\Phi_0 - I\| \leq 2\|H_0\| \leq 2\phi\|\mathbf{c}^0 - \mathbf{c}^*\| \leq \frac{\xi}{4}, \tag{68}$$

$$\|V_1 - V_0\| \leq \|\Psi_0 - I\| \leq 2\|K_0\| \leq 2\phi\|\mathbf{c}^0 - \mathbf{c}^*\| \leq \frac{\xi}{4}. \tag{69}$$

Now, we verify that (50) holds for $k = 1$. From (58) and (68)–(69),

$$\|E_1^{(1)}\| = \|U_{11}^{(1)} - U_{11}(\mathbf{c}^*)\| \leq \|U_1 - U_0\| + \|U_{11}^{(0)} - U_{11}(\mathbf{c}^*)\| \leq \frac{\xi}{4} + \frac{\xi}{2} \leq \xi,$$

$$\|E_2^{(1)}\| = \|V_1 - V(\mathbf{c}^*)\| \leq \|V_1 - V_0\| + \|V_0 - V(\mathbf{c}^*)\| \leq \frac{\xi}{4} + \frac{\xi}{2} \leq \xi.$$

Next, we show that (51) holds for $k = 1$. By (48), (53), and (66)–(67),

$$\begin{aligned}&\max\{\|\mathbf{c}^2 - \mathbf{c}^1\|, \|\boldsymbol{\sigma}^1 - \boldsymbol{\sigma}^*\|, \|H_1\|, \|K_1\|, \|U_2 - U_1\|, \|V_2 - V_1\|\} \\ &\leq \phi\max\{\|\boldsymbol{\sigma}^0 - \boldsymbol{\sigma}^*\|^\beta, \|H_0\|^\beta, \|K_0\|^\beta\} \leq \phi^{1+\beta}\|\mathbf{c}^0 - \mathbf{c}^*\|^\beta.\end{aligned} \tag{70}$$

If we let $\varphi \equiv \phi^{\frac{\beta}{\beta-1}}\epsilon$, then by (49), we have

$$\varphi^{\beta} \leq \epsilon < 1. \tag{71}$$

Thus by (70),

$$\begin{aligned}
&\max\{\|\mathbf{c}^2-\mathbf{c}^1\|, \|\boldsymbol{\sigma}^1-\boldsymbol{\sigma}^*\|, \|H_1\|, \|K_1\|, \|U_2-U_1\|, \|V_2-V_1\|\} \\
&\leq \phi^{1+\beta}\|\mathbf{c}^0-\mathbf{c}^*\|^{\beta} = (\phi^{\frac{1+\beta}{\beta}}\|\mathbf{c}^0-\mathbf{c}^*\|)^{\beta} \leq (\phi^{\frac{1+\beta}{\beta}}\epsilon)^{\beta} \leq \varphi^{\beta} \leq \epsilon.
\end{aligned}$$

That is, (52) holds for $k=1$.

Now, we assume that (50)–(52) hold for all positive integer less than or equal to $k-1$. We first prove that (50) holds for k. In fact, by (51), we have, for $j = 1, 2, \ldots, k-1$,

$$\begin{aligned}
\max\{\|U_{j+1}-U_j\|, \|V_{j+1}-V_j\|\} &\leq \phi^{1+\beta+\cdots+\beta^j}\|\mathbf{c}^0-\mathbf{c}^*\|^{\beta^j}, \\
&\leq \left(\phi^{\frac{1+\beta+\cdots+\beta^j}{\beta^j}}\|\mathbf{c}^0-\mathbf{c}^*\|\right)^{\beta^j} \\
&= \left(\phi^{\left(\frac{1}{\beta^j}+\frac{1}{\beta^{j-1}}+\cdots+1\right)}\|\mathbf{c}^0-\mathbf{c}^*\|\right)^{\beta^j} \\
&\leq (\phi^{\frac{\beta}{\beta-1}}\|\mathbf{c}^0-\mathbf{c}^*\|)^{\beta^j} \leq \varphi^{\beta^j}.
\end{aligned} \tag{72}$$

Then, by (72), (69), (58), and (71), we get

$$\begin{aligned}
\|V_k - V(\mathbf{c}^*)\| &\leq \sum_{j=1}^{k-1}\|V_{j+1}-V_j\| + \|V_1-V_0\| + \|V_0-V(\mathbf{c}^*)\| \\
&\leq \sum_{j=1}^{k-1}\varphi^{\beta^j} + \frac{\xi}{4} + \frac{\xi}{2} \leq \sum_{j=1}^{k-1}\varphi^{1+j\ln\beta} + \frac{\xi}{4} + \frac{\xi}{2} \\
&\leq \sum_{j=1}^{k-1}(\varphi^{\ln\beta})^j + \frac{\xi}{4} + \frac{\xi}{2} \leq \frac{\varphi^{\ln\beta}}{1-\varphi^{\ln\beta}} + \frac{\xi}{4} + \frac{\xi}{2} \\
&\leq \frac{\epsilon^{\frac{\ln\beta}{\beta}}}{1-\epsilon^{\frac{\ln\beta}{\beta}}} + \frac{\xi}{4} + \frac{\xi}{2} \leq \frac{\xi}{4} + \frac{\xi}{4} + \frac{\xi}{2} = \xi.
\end{aligned}$$

By the same argument, we can prove that $\|U_{11}^{(k)} - U_{11}(\mathbf{c}^*)\| \leq \xi$. Therefore, (50) holds for k.

To prove that (51) holds for k, we use (48):

$$\begin{aligned}
&\max\{\|\mathbf{c}^{k+1}-\mathbf{c}^k\|, \|\boldsymbol{\sigma}^k-\boldsymbol{\sigma}^*\|, \|H_k\|, \|K_k\|, \|U_{k+1}-U_k\|, \|V_{k+1}-V_k\|\} \\
&\leq \phi\max\{\|\boldsymbol{\sigma}^{k-1}-\boldsymbol{\sigma}^*\|^{\beta}, \|H_{k-1}\|^{\beta}, \|K_{k-1}\|^{\beta}\}, \\
&\leq \phi\big(\phi^{1+\beta+\cdots+\beta^{k-1}}\|\mathbf{c}^0-\mathbf{c}^*\|^{\beta^{k-1}}\big)^{\beta}, \\
&= \phi^{1+\beta+\cdots+\beta^k}\|\mathbf{c}^0-\mathbf{c}^*\|^{\beta^k}.
\end{aligned} \tag{73}$$

To verify that (52) holds for k. By (73), we have

$$
\begin{aligned}
&\max\{\|\mathbf{c}^{k+1}-\mathbf{c}^k\|, \|\boldsymbol{\sigma}^k-\boldsymbol{\sigma}^*\|, \|H_k\|, \|K_k\|, \|U_{k+1}-U_k\|, \|V_{k+1}-V_k\|\} \\
&\le \left(\phi^{\frac{1+\beta+\cdots+\beta^k}{\beta^k}}\|\mathbf{c}^0-\mathbf{c}^*\|\right)^{\beta^k} = \left(\phi^{\left(\frac{1}{\beta^k}+\frac{1}{\beta^{k-1}}+\cdots+1\right)}\|\mathbf{c}^0-\mathbf{c}^*\|\right)^{\beta^k} \\
&\le (\phi^{\frac{\beta}{\beta-1}}\|\mathbf{c}^0-\mathbf{c}^*\|)^{\beta^k} \le \varphi^{\beta^k} \qquad (74)\\
&\le \varphi^{\beta} \le \epsilon.
\end{aligned}
$$

Thus we have proved that (50)–(52) hold for any $k \ge 1$.

Finally, we show the local convergence of the sequence $\{\mathbf{c}^k\}$. From (74), we have, for any integer $m \ge 1$,

$$
\begin{aligned}
\left\|\mathbf{c}^{k+m}-\mathbf{c}^k\right\| &\le \sum_{l=1}^{m}\left\|\mathbf{c}^{k+l}-\mathbf{c}^{k+l-1}\right\| \le \sum_{l=1}^{m}\varphi^{\beta^{k+l-1}} = \sum_{l=1}^{m}\left(\varphi^{\beta^{k-1}}\right)^{\beta^l} \le \sum_{l=1}^{m}\left(\varphi^{\beta^{k-1}}\right)^{1+l\ln\beta} \\
&\le \sum_{l=1}^{m}\left(\varphi^{\beta^{k-1}\ln\beta}\right)^l \le \frac{\varphi^{\beta^{k-1}\ln\beta}-\left(\varphi^{\beta^{k-1}\ln\beta}\right)^{m+1}}{1-\varphi^{\beta^{k-1}\ln\beta}}. \qquad (75)
\end{aligned}
$$

Similarly, we can show that

$$
\begin{aligned}
\left\|U_{k+m}-U_k\right\| &\le \frac{\varphi^{\beta^{k-1}\ln\beta}-\left(\varphi^{\beta^{k-1}\ln\beta}\right)^{m+1}}{1-\varphi^{\beta^{k-1}\ln\beta}}, \\
\left\|V_{k+m}-V_k\right\| &\le \frac{\varphi^{\beta^{k-1}\ln\beta}-\left(\varphi^{\beta^{k-1}\ln\beta}\right)^{m+1}}{1-\varphi^{\beta^{k-1}\ln\beta}},
\end{aligned}
$$

This shows that $\{\mathbf{c}^k\}$, $\{U_k\}$, and $\{V_k\}$ are Cauchy sequences since $\varphi \le \epsilon^{\frac{1}{\beta}} < 1$. Therefore, there exists a vector $\mathbf{c}^\dagger \in \mathbb{R}^n$, two matrices $U_\dagger \in \mathscr{O}(m)$ and $V_\dagger \in \mathscr{O}(n)$ such that $\mathbf{c}^k \to \mathbf{c}^\dagger$, $U_k \to U_\dagger$, and $V_k \to V_\dagger$ as $k \to \infty$.

□

Remark 4.11 *We remark that $\mathbf{c}^\dagger$ may not equal to the original solution $\mathbf{c}^*$, and $U_\dagger$ and $V_\dagger$ is not necessarily the same as $U(\mathbf{c}^*)$ and $V(\mathbf{c}^*)$, respectively, see for instance [4].*

We end this section by establishing the root convergence rate of our method. Based on the estimate of the sequence $\{\mathbf{c}^k\}$ in (75), we can directly use the definition of the root convergence to derive the rate of the sequence $\{\mathbf{c}^k\}$ as follows.

Theorem 4.12 *Under the same conditions as in Theorem 4.10, the sequence $\{\mathbf{c}^k\}$ converges to the limit $\mathbf{c}^\dagger$ with root-convergence rate at least equal to β.*

Proof: By Theorem 4.10, we know that $\{\mathbf{c}^k\}$ is locally convergent with

$$\lim_{k\to\infty}\mathbf{c}^k = \mathbf{c}^\dagger.$$

Since $\varphi < 1$, let $m \to \infty$ in (75), we have, for each $k \ge 1$,

$$\left\|\mathbf{c}^k-\mathbf{c}^\dagger\right\| \le \frac{\varphi^{\beta^{k-1}\ln\beta}}{1-\varphi^{\beta^{k-1}\ln\beta}} \le \omega(\varphi^{\ln\beta})^{\beta^{k-1}},$$

where $\omega = \frac{1}{1-\varphi^{\ln\beta}} > 1$.

We now estimate the root-convergence factors of $\{\mathbf{c}^k\}$ defined in (42) for different values of p:

1. If $p = 1$, then

$$R_1\{\mathbf{c}^k\} = \limsup_{k\to\infty} \|\mathbf{c}^k - \mathbf{c}^\dagger\|^{1/k} \le \limsup_{k\to\infty} \omega^{1/k}\left(\varphi^{\ln\beta/\beta}\right)^{\beta^k/k} = 0.$$

2. If $1 < p < \beta$, then

$$R_p\{\mathbf{c}^k\} = \limsup_{k\to\infty} \|\mathbf{c}^k - \mathbf{c}^\dagger\|^{1/p^k} \le \limsup_{k\to\infty} \omega^{1/p^k}(\varphi^{\ln\beta/\beta})^{(\beta/p)^k} = 0.$$

3. If $p = \beta$, then

$$R_p\{\mathbf{c}^k\} = \limsup_{k\to\infty} \|\mathbf{c}^k - \mathbf{c}^\dagger\|^{1/\beta^k} \le \limsup_{k\to\infty} \omega^{1/p^k}\varphi^{\ln\beta/\beta} = \varphi^{\ln\beta/\beta} < 1.$$

4. If $p > \beta$, then

$$R_p\{\mathbf{c}^k\} = \limsup_{k\to\infty} \|\mathbf{c}^k - \mathbf{c}^\dagger\|^{1/p^k} \le \limsup_{k\to\infty} \omega^{1/p^k}(\varphi^{\ln\beta/\beta})^{(\beta/p)^k} = 1.$$

Therefore, $R_p\{\mathbf{c}^k\} = 0$ for any $p \in [1, \beta)$ and $R_p\{\mathbf{c}^k\} \le 1$ for any $p \in [\beta, \infty)$. Thus according to (43), $O_R(\mathbf{c}^\dagger) \ge \beta$.

□

Remark 4.13 *From the proof of Theorem 4.12, we easily see that under the same assumptions as in Theorem 4.10, the sequences $\{\boldsymbol{\sigma}^k\}$, $\{U_k\}$ and $\{V_k\}$ converge to their limit $\boldsymbol{\sigma}^*$, $U_\dagger$ and $V_\dagger$ with a convergence rate at least equal to β in the root sense, respectively.*

5 Numerical Tests

In this section, we compare the numerical performance of Algorithm II with that of Algorithm I. Our goal is to illustrate the advantage of our method over Algorithm I in terms of the reduction of oversolving problem and the overall computational cost.

The test were carried out in `Matlab` 7.0.4 running on a PC Inter Pentium R of 3.00 GHz CPU. All the basis matrices $\{A_i\}_{i=0}^n$ defined in (1) were generated randomly by `Matlab`-provided `randn` function.

For demonstration purposes, we consider the following three problem sizes: `a)` m=100 and n=60; `b)` m=150 and n=100; `c)` m=300 and n=200. For the given dimensions m and n, we first randomly generate a vector $\mathbf{c}^* \in \mathbb{R}^n$ and compute the singular values $\{\sigma_i^*\}_{i=1}^n$ of $A(\mathbf{c}^*)$ as the prescribed singular values. Since both of the algorithms are proved to converge locally, we form the initial guess $\mathbf{c}^0$ by chopping the components of $\mathbf{c}^*$ to three decimal places for the cases of `a)` and `b)` and to four decimal places for the case of `c)`.

The six linear systems (7), (13)–(14) of algorithm I, and (16), (18)–(19) of Algorithm II were all solved by the QMR method [23] via the `Matlab`-provided `QMR` function. `QMR(A,b,tol,itmax,P1,P2,x0)` attempts to solve the system of linear equations $A\mathbf{x} = \mathbf{b}$

for $\mathbf{x}$, where `tol`, `itmax`, `P1` and `P2`, and $\mathbf{x}_0$ denote the stop tolerance, the maximum number of iterations, the preconditioners, and the initial guess, respectively. For the system (16), we solve the linear system

$$J_k \mathbf{c}^{k+1} = \boldsymbol{\sigma}^* - \mathbf{a}^k$$

by the QMR method with the stop tolerance `tol` being $\|\mathbf{r}^k\|$ defined by (17). To guarantee the orthogonality of the iterates P_k and Q_k and U_k and V_k, equations (13)–(14) and (18)–(19) are solved iteratively up to the machine precision ($\approx 2.22 \times 10^{-16}$). To speed up the convergence, one may use the preconditioned QMR method for equations (7) and (16). Here, we use `Matlab`-provided ILU (Incomplete LU factorization) preconditioner: `LUINC(A,drop-tolerance)` since the ILU preconditioner is one of the most versatile preconditioners for unstructured matrices [12, 18]. For all the three problem sizes, we set the drop tolerance to be 0.01. Also, we remark that we only try to illustrate that preconditioning can be incorporated easily. We are not attempting to find the best preconditioners for these systems.

We use $\mathbf{c}^k$, the iterant at the kth Newton iteration, as the initial guess for solving the approximate Jacobian equations (7) and (16) iteratively in the $(k+1)$th outer iteration. In Algorithm II, the stopping tolerance for (16) is given in (17). Since Algorithm I is the exact version of Algorithm II, (7) should be solved iteratively up to the machine precision. Here, we try to use a large value 10^{-14} as the stopping tolerance for (7), and compare the two algorithms. When the problem size is larger, the stopping tolerance for (7) should be smaller or even up to the machine precision so that the outer iteration converges. The outer iterations of Algorithms I and II were stopped, respectively, when

$$\|P_k^T A(\mathbf{c}^k) Q_k - \Sigma_*\|_F \leq 10^{-10} \quad \text{and} \quad \|U_k^T A(\mathbf{c}^k) V_k - \Sigma_*\|_F \leq 10^{-10}.$$

We now report our experimental results. Table 1 lists the total numbers of outer iterations N_o averaged over ten tests, the average total numbers of inner iterations N_i required for solving the Jacobian equations in Algorithms I and II. In particular, 'I' and 'P' denote no preconditioner or the ILU preconditioner are used, respectively. For simplicity of demonstration, for Case `c)`, we only display the numbers of outer and inner iterations with the ILU-preconditioner used. From Table 1 we see that N_o is small for Algorithm I and also for Algorithm II when $\beta \geq 1.5$. This confirms the theoretical convergence rate of the two algorithms. However, under small N_0, Algorithm II for $\beta \approx 1.5$ is more efficient than Algorithm I with respect to N_i. We also note that the ILU preconditioner is effective for the Jacobian systems.

We point out that there is no theoretical guarantee for the selection of the optimal value of β though we have showed the superlinear convergence of our method. In practice, we may first try small β (e.g. between 1.5 and 1.6) while preserving the small out iterations. In addition, though the actual reduction of inner iteration numbers is not so much (e.g. the numbers of inner iterations vary from around 25.5 to around 13.3 in Case a) and from around 43.4 to around 23.0 in Case c)), our method have improved the computing speed very fast over the exact version. According to the well-known Moore's law, the computing speed of machines would double periodically approximately every 18 months. Compared with this, our method is already very efficient as expected.

To further illustrate the oversolving problem, we investigate the convergence history of Algorithms I and II for one of the tests when $m = 100$ and $n = 60$. Specifically, at each inner iteration, we computed the the Euclidean vector norm error e between the current approximation and its limits point $\mathbf{c}^\dagger$ (not necessarily the same as $\mathbf{c}^*$). Figure 1 depicts

Table 1: Averaged total numbers of outer and inner iterations.

n			Alg. I	β in Alg. II									
				1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0
`a)`	I	N_o	3.6	9.3	5.4	4.6	3.8	3.6	3.6	3.6	3.6	3.6	3.6
		N_i	411	611	394	358	322	312	332	335	354	367	375
	P	N_o	3.6	6.8	5.0	4.2	3.6	3.6	3.6	3.6	3.6	3.6	3.6
		N_i	25.5	11.6	11.8	12.6	13.3	15.2	17.0	19.5	21.5	22.1	23.7
`b)`	I	N_o	3.6	10.4	6.0	4.8	4.0	3.6	3.6	3.6	3.6	3.6	3.6
		N_i	966	1650	1066	876	794	751	778	834	839	861	922
	P	N_o	3	7.0	5.4	4.4	3.8	3.6	3.6	3.6	3.6	3.6	3.6
		N_i	30.8	13.2	14.6	15.6	17.2	19.2	20.6	22.6	24.6	28.0	28.4
`c)`	P	N_o	3	7.5	4.6	3.6	3	3	3	3	3	3	3
		N_i	43.4	20.0	24.0	21.5	23.0	26.6	27.0	29.0	33.0	34.4	36.6

the logarithm of e versus the number of inner iterations for Algorithm I and Algorithm II with $\beta = 1.5$ and 2. We also mark the error at the outer iterations with special symbols. We observe from Figure 1 that our method converges faster than Algorithm I. Moreover, there is a significant oversolving problem for Algorithm I (see the horizontal lines between iteration numbers 77 to 158 and 168 to 273) whereas there are almost no oversolving for Algorithm II with $\beta = 1.5$.

To show that it requires only a few iterations for solving the linear equations (13)–(14) and (18)– (19), we display in Table 2 the numbers of iterations required for convergence for these systems, averaged over the ten test problems for the cases of `a)` and `b)`. We observe from Table 2 that the number of inner iterations required is small and decreases as the outer iteration progresses. Thus one can confidently solve these systems by iterative solvers without any preconditioning.

Next, we test one of the tests with the problem size $m = 100$ and $n = 60$ for varying initial guesses $\mathbf{c}^0$: the initial guess $\mathbf{c}^0$ was formed by chopping the components of $\mathbf{c}^*$ to 1) three decimal places; 2) two decimal places; 3) one decimal places. The numerical results are listed in Tables 3 and 4, where `Init.`, `No.`, `Ni.`, `Res0.`, `Res*.`, `Tol0.`, and `Tol*.` stand for, respectively, the initial point used, the number of outer iterations, the number of inner iterations, the residual $\|\mathbf{c}^k - \mathbf{c}^*\|$ at the starting point $\mathbf{c}^0$ and the iterate when an algorithm converges, the residual $\|\boldsymbol{\sigma}(\mathbf{c}^k) - \boldsymbol{\sigma}^*\|$ at the starting point $\mathbf{c}^0$ and the iterate when an algorithm converges (we set the maximal out iterations to be 20). Table 4 displays the residual $\|\boldsymbol{\sigma}(\mathbf{c}^k) - \boldsymbol{\sigma}^*\|$ for Cases 1) and 2) throughout the outer iterations. It is obvious that the superlinear convergence occurs in practice. Also, Table 3 implies that a starting point $\mathbf{c}^0$ should be in the domain of the convergence of an algorithm. Otherwise, the algorithm will be out of convergence. A friendly initial guess can be found by the global strategy as noted in Section 1.

Finally, we remark that we also retried the same tests with different iterative solvers (e.g. BiCG [26] and CGS [24]) together with their ILU-preconditioned versions for solving the Jacobian equations. All these solvers behavior similar as the QMR method and its ILU-preconditioned version. Clearly, the ILU-preconditioner is not the best one. In general, the Jacobian matrices ((7) and (16)) are nonsymmetric and dense, it needs further study to find a better preconditioner, see for instance [5] for this topic.

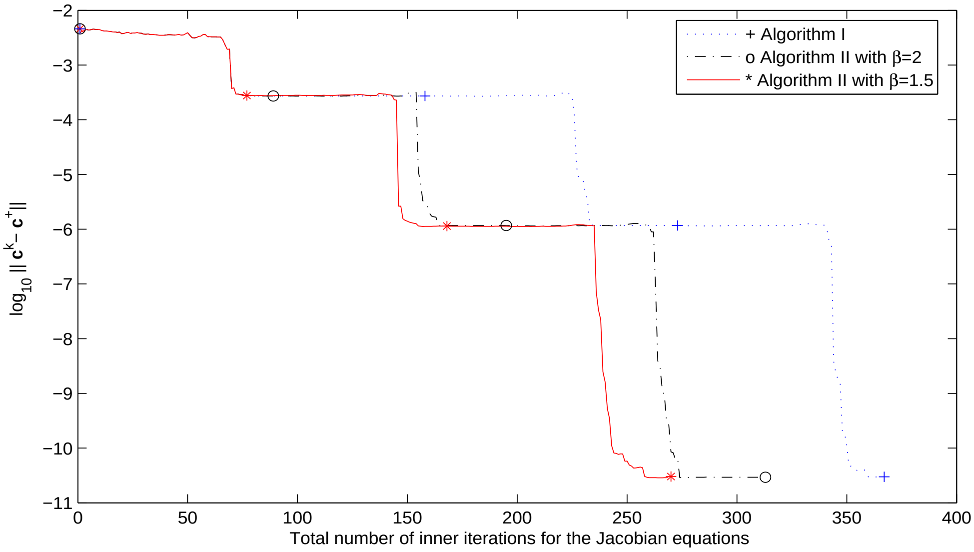

Figure 1: Convergence history of one of the tests.

Table 2: Averaged total numbers of inner iterations in Step (f) of Algorithms I and II.

	a)			b)		
Outer iteration	1st	2nd	3rd	1st	2nd	3rd
Alg. I	12.0	7.5	4.0	13.5	8.5	4.0
Alg. II with $\beta = 2.0$	14.0	8.0	4.0	14.0	8.0	4.0
Alg. II with $\beta = 1.5$	12.5	7.7	4.0	14.0	8.0	4.0

6 Concluding Remarks

In this paper, we have proposed an inexact Newton-type method for the inverse singular value problem. We show that our inexact method converges superlinearly. The inexact version can minimize the oversolving problem of the Newton-type method and give a good tradeoff between the inner and outer iterations. We also present numerical experiments to illustrate our results.

Further work includes whether the proposed inexact Newton-type method can be extended to the ill-posed case, i.e., the given singular values affected by noise.

Acknowledgments. We would like to thank Prof. Raymond Chan for his valuable discussion and comments when we began this paper. This work was partially supported by the HKRGC Grant CUHK 400503.

References

[1] E. L. Allgower and K. Georg, *Continuation and path following*, Acta Numer., 2 (1993), pp. 1–64.

Table 3: Numerical results for different initial guesses.

Init.	Algorithm	No.	Ni.	Res0.	Res*.	Tol0.	Tol*.
1)	Alg. I	3	380	4.6×10^{-3}	3.2×10^{-11}	4.1×10^{-2}	1.1×10^{-11}
	Alg. II ($\beta = 2.0$)	3	343	4.6×10^{-3}	3.2×10^{-11}	4.1×10^{-2}	1.2×10^{-11}
	Alg. II ($\beta = 1.5$)	3	247	4.6×10^{-3}	3.8×10^{-11}	4.1×10^{-2}	2.5×10^{-11}
2)	Alg. I	5	565	4.6×10^{-2}	4.0×10^{-12}	3.3×10^{-1}	4.1×10^{-12}
	Alg. II ($\beta = 2.0$)	5	498	4.6×10^{-2}	3.9×10^{-12}	3.3×10^{-1}	8.5×10^{-13}
	Alg. II ($\beta = 1.5$)	5	424	4.6×10^{-2}	4.0×10^{-12}	3.3×10^{-1}	8.1×10^{-13}
3)	Alg. I	≥ 20	*	*	*	*	*
	Alg. II ($\beta = 2.0$)	≥ 20	*	*	*	*	*
	Alg. II ($\beta = 1.5$)	≥ 20	*	*	*	*	*

Table 4: Errors of singular values.

No.	Case 1)			Case 2)		
	Alg. I	Alg. II		Alg. I	Alg. II	
		$\beta = 2.0$	$\beta = 1.5$		$\beta = 2.0$	$\beta = 1.5$
0	4.1×10^{-2}	4.1×10^{-2}	4.1×10^{-2}	3.3×10^{-1}	3.2×10^{-1}	3.3×10^{-1}
1	2.0×10^{-4}	2.0×10^{-4}	3.8×10^{-4}	2.3×10^{-2}	2.3×10^{-2}	2.5×10^{-2}
2	6.3×10^{-7}	6.3×10^{-7}	7.3×10^{-7}	1.6×10^{-3}	1.6×10^{-3}	1.7×10^{-3}
3	1.1×10^{-11}	1.1×10^{-11}	2.5×10^{-11}	2.4×10^{-5}	2.4×10^{-5}	2.7×10^{-5}
4				3.5×10^{-8}	3.4×10^{-8}	4.6×10^{-8}
5				4.1×10^{-12}	8.5×10^{-13}	8.1×10^{-13}

[2] Z.-J. Bai, *Inexact Newton methods for inverse eigenvalue problems*, Appl. Math. Comput., 172 (2006), pp. 682–689.

[3] Z.-J. Bai, R. H. Chan, and B. Morini, *An inexact Cayley transform method for inverse eigenvalue problems*, Inverse Problems, 20 (2004), pp. 1675–1689.

[4] Z.-J. Bai, B. Morini, and S. F. Xu, *On the local convergence of an iterative approach for inverse singular value problems*, J. Comput. Appl. Math., 198 (2007), pp. 344–360.

[5] M. Benzi, *Preconditioning techniques for large linear systems: A survey*, J. Comput. Phys., 182 (2002), pp. 418-477.

[6] R. H. Chan, H. L. Chung, and S. F. Xu, *The inexact Newton-like method for inverse eigenvalue problem*, BIT, 43 (2003), pp. 7–20.

[7] D Chu and M. T. Chu, *Low rank update of singular values*, Math. Comp., 75 (2006), pp. 1351–1366.

[8] M. T. Chu, *Numerical methods for inverse singular value problems*, SIAM J. Numer. Anal., 29 (1992), pp. 885–903.

[9] M. T. Chu, *Inverse eigenvalue problems*, SIAM Rev., 40 (1998), pp. 1–39.

[10] M. T. Chu and G. H. Golub, *Structured inverse eigenvalue problems*, Acta Numer., 11 (2002), pp. 1–71.

[11] M. T. Chu and G. H. Golub, *Inverse Eigenvalue Problems: Theory, Algorithms, and Applications*, Oxford University Press, 2005.

[12] T. Dupont, R. P. Kendall, and H. H. JR. Rachford, *An approximate factorization procedure for solving self-adjoint elliptic difference equations*, SIAM J. Numer. Anal., 5 (1968), pp. 559–573.

[13] S. C. Eisenstat and H. F. Walker, *Choosing the forcing terms in an inexact Newton method*, SIAM J. Sci. Comput., 17 (1996), pp. 16–32.

[14] D. R. Fokkema, G. L. G. Sleijen, and H. A. van der Vorst, *Accelerated inexact Newton schemes for large systems of nonlinear equations*, SIAM J. Sci. Comput., 19 (1998), pp. 657–674.

[15] S. Friedland, J. Nocedal, and M. L. Overton, *The formulation and analysis of numerical methods for inverse eigenvalue problems*, SIAM J. Numer. Anal., 24 (1987), pp. 634–667.

[16] A. G. Gramm, *Inverse Problems: Mathematical and Analytical Techniques with Applications to Engineering*, Springer, 2005.

[17] G. H. Golub and C. F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, 1996.

[18] I. Gustafsson, *A class of first order factorizations*, BIT, 18 (1978), pp. 142–156.

[19] M. R. Hestenes and E. L. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Res. Nat. Bur. Standards,, 49 (1952), pp. 409–436.

[20] L. P. Lebedev, I. I. Vorovich, and G. M. Gladwell, *Functional Analysis: Applications in Mechanics and Inverse Problems*, Kluwer Academic Publishers, 2002.

[21] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, 1970.

[22] T Politi, *A discrete approach for the inverse singular value problem in some quadratic group*, Lecture Notes in Comput. Sci., 2658 (2003), pp. 121–130.

[23] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, 1996.

[24] P. Sonneveld, *CGS: A fast Lanczos-type solver for nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 10 (1989), pp. 36–52.

[25] G. Uhlmann and S. Levy (Editors): *Inside Out: Inverse Problems and Applications*, Cambridge University Press, 2003.

[26] H. A. van der Vorst, *BiCGSTAB: A fast and smoothly converging variant of the Bi-CG for the solution of nonsymmetric linear systems*, SIAM J. Sci. and Stat. Comp., 13 (1992), pp. 631–644.

[27] S. F. Xu, *An Introduction to Inverse Eigenvalue Problems*, Peking University Press, 1998.