An Inexact Intercept Method for Inverse Singular Value Problems

Zheng-Jian Bai*

October 24, 2006

Abstract

The intercept method is a Newton-type iterative approach for solving inverse singular value problems originally given by M. Chu [SIAM J. Numer. Anal., 29 (1992), pp. 885–903]. When the problem is large, one may employ iterative methods to solve the Jacobian equations. However, iterative methods usually cause the oversolving problem in the sense that they require far more (inner) iterations than is required for the convergence of the Newton (outer) iterations. In this paper, we propose an inexact intercept method which can reduce or even minimize the oversolving problem and improve the efficiency with respect to the exact version. Our inexact iterative approach is proved to converge superlinearly and a good tradeoff between the required inner and outer iterations can be reached.

Keywords. Inverse problem, singular value, root-convergence rate

AMS subject classifications. 15A29, 15A18, 65F15, 65F18

1 Introduction

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

\[ A(c) \equiv A_0 + \sum_{i=1}^n c_i A_i, \]

and denote the singular values of \( A(c) \) by \( \{\sigma_i(c)\}_{i=1}^n \), where \( \sigma_1(c) \geq \sigma_2(c) \geq \cdots \geq \sigma_n(c) \geq 0 \). 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 \( c \in \mathbb{R}^n \) such that \( \sigma_i(c) = \sigma_i^* \) for \( i = 1, 2, \ldots, n \). The problem was first proposed by Chu [4].

This a special class of parameterized ISVPs. General ISVPs concern the construction of a structured matrix satisfying the given singular values. Naturally, ISVPs can be seen as the extension of the inverse eigenvalue problems (IEPs). For the applications, existence theory, and numerical methods of the IEPs, we can refer to the survey papers [5, 6] and the books [7, 18]. The ISVP can be converted into an IEP, see for instance [4]. However, the additional conditions make the solvability of the corresponding IEP more complicated and difficult. Likewise, the existence question for the ISVP should be an interesting topic for further research.

*Department of Information and Computational Mathematics, Xiamen University, Xiamen 361005, People’s Republic of China (zjbai@xmu.edu.cn). The research was partially supported by the National Natural Science Foundation of China Grant 10601043 and Xiamen University Grant 0000-X07152.
We note that the ISVP can be formulated as a system of nonlinear equations
\[ f(c) = (\sigma_1(c) - \sigma_*^1, \ldots, \sigma_n(c) - \sigma_*^n)^T = 0. \] (2)
The second method in [4], called the intercept method, may be seen as a Newton-type method applied the nonlinear system (2). In each Newton (outer) iteration, we need to solve an approximate Jacobian equation. If the dimension \( n \) is large, directly solving such a linear system will be very expensive. The cost can be reduced 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 [9]. The inexact Newton-type method aims to stop the inner iteration before convergence. By choosing suitable stopping criteria, we can reduce the total cost of the whole inner-outer iterations. In fact, the approximate Jacobian equation may not be solved exactly in order that the Newton method converges fast.

In this paper, we propose an inexact version of the intercept method. Our inexact method solves the approximate Jacobian equation inexacty by stopping the inner iterations before convergence. We provide a new stopping tolerance to stop the inner iterations at each Newton (outer) iteration. The theoretical and experimental results are presented. We will show that our method converges superlinearly. Moreover, our method can nearly avoid the oversolving problem which is illustrated by our numerical examples.

We note that our inexact method is just an inexact Newton-type method which converges fast but locally. In practice, we should combine the continuous method (e.g. the continuous method in [4, Section 2]), which converges globally but slowly, with our inexact iterative method. We first use the continuous method as the predictor step to find a start point for our iterative method which, then, improves the accuracy, see for instance [1].

This paper is outlined as follows. In Section 2 we recall the intercept method for solving the ISVP. In Section 3 we introduce our inexact version. In Section 4 we give the convergence analysis of our method. Finally, in Section 5 we report some numerical examples to illustrate our results.

2 The Intercept Method

In this section, we briefly recall the intercept method proposed by Chu [4]. For simplicity, we assume that all the singular values \( \{\sigma_i^*\}_{i=1}^n \) are positive and distinct. In what follows, let \( \| \cdot \| \) denote the Euclidean vector norm or its corresponding induced matrix norm. \( I \) is the identity matrix. Let \( \Sigma_* = \text{diag}(\sigma_1^*, \ldots, \sigma_n^*) \in \mathbb{R}^{m \times n} \) and \( \mathcal{O}(n) \) denote the set of all orthogonal matrices in \( \mathbb{R}^{n \times n} \). Define the affine subspace \( \mathcal{A} \equiv \{ A(c) | c \in \mathbb{R}^n \} \) and the surface \( \mathcal{G}_s(\Sigma_*) = \{ P\Sigma_*Q^T | P \in \mathcal{O}(m), Q \in \mathcal{O}(n) \} \), i.e. the set of all matrices in \( \mathbb{R}^{m \times n} \) with singular values \( \sigma_1^* > \sigma_2^* \cdots > \sigma_n^* > 0 \). Then, solving the ISVP is equivalent to finding an intersection of \( \mathcal{G}_s(\Sigma_*) \) and \( \mathcal{A} \).

The intercept approach in [2] is simply a generalization of Method III in [11]. Let \( c^* \) be a solution of the ISVP. Suppose that \( Y_k \in \mathcal{G}_s(\Sigma_*) \), there exist \( P_k \in \mathcal{O}(n) \) and \( Q_k \in \mathcal{O}(n) \) such that
\[ Y_k = P_k\Sigma_*Q_k^T. \] (3)
The new iterate \( c^{k+1} \in \mathbb{R}^n \) is determined by seeking a \( \mathcal{A} \)-intercept \( A(c^{k+1}) \) from a line that is tangent to the manifold \( \mathcal{G}_s(\Sigma_*) \) 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 $c^{k+1} \in \mathbb{R}^n$ such that

$$Y_k + C_kY_k - Y_kD_k = A(c^{k+1}).$$

By (3), we have

$$\Sigma_x + \tilde{C}_k\Sigma_x - \Sigma_x\tilde{D}_k = P_k^T A(c^{k+1})Q_k \equiv X_k,$$  \hspace{1cm} (4)

where $\tilde{C}_k = P_k^T C_kP_k$ and $\tilde{D}_k = Q_k^T D_kQ_k$ are two skew-symmetric matrices.

We observe from (4) that the lower-right corner of size $(m - n)$-by-$(m - n)$ in $\tilde{C}_k$ can be arbitrary. In [4], these entries are set to be identically zeros, i.e.,

$$[\tilde{C}_k]_{ij} = 0 \quad \text{for } n + 1 \leq i \neq j \leq m.$$  \hspace{1cm} (5)

In fact, different allocations of these free entries have a direct effect on the convergence speed of the intercept method, see [2].

For $1 \leq i = j \leq n$, by (4), we obtain the following equation

$$J_kc^{k+1} = \sigma^* - a^k,$$  \hspace{1cm} (6)

where

$$[J_k]_{ij} \equiv (p^k_i)^T A_j q^k_i, \quad 1 \leq i, j \leq n,$$  \hspace{1cm} (7)

$$\sigma^* \equiv (\sigma^*_1, \ldots, \sigma^*_n)^T,$$  \hspace{1cm} (8)

$$a^k_i \equiv (p^k_i)^T A_0 q^k_i, \quad 1 \leq i \leq n,$$  \hspace{1cm} (9)

where $p^k_i$ and $q^k_i$ are the column vectors of $P_k$ and $Q_k$, respectively. If the matrix $J_k$ is nonsingular, then we can solve (6) for the vector $c^{k+1}$.

Next, the skew-symmetric matrices $\tilde{C}_k$ and $\tilde{D}_k$ are obtained by comparing the “off-diagonal” entries in (4). For $n + 1 \leq i \leq m, 1 \leq j \leq n$, we have

$$[\tilde{C}_k]_{ij} = -[\tilde{C}_k]_{ji} = \frac{[X_k]_{ij}}{\sigma^*_j}.$$  \hspace{1cm} (10)

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

$$[X_k]_{ij} = \sigma^*_j[\tilde{C}_k]_{ij} - \sigma^*_i[\tilde{D}_k]_{ij},$$

$$[X_k]_{ji} = \sigma^*_i[\tilde{C}_k]_{ji} - \sigma^*_j[\tilde{D}_k]_{ji} = -\sigma^*_i[\tilde{C}_k]_{ij} + \sigma^*_j[\tilde{D}_k]_{ij},$$

which lead to

$$[\tilde{C}_k]_{ij} = -[\tilde{C}_k]_{ji} = \frac{\sigma^*_j[X_k]_{ji} + \sigma^*_i[X_k]_{ij}}{(\sigma^*_j)^2 - (\sigma^*_i)^2},$$  \hspace{1cm} (10)

$$[\tilde{D}_k]_{ij} = -[\tilde{D}_k]_{ji} = \frac{\sigma^*_i[X_k]_{ij} + \sigma^*_j[X_k]_{ji}}{(\sigma^*_j)^2 - (\sigma^*_i)^2}. $$  \hspace{1cm} (11)

This completes the intercept step.

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

$$Y_{k+1} \equiv P_{k+1}\Sigma_xQ^T_{k+1},$$

3
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
\[ S_k \equiv \left( I + \frac{1}{2} \tilde{C}_k \right) \left( I - \frac{1}{2} \tilde{C}_k \right)^{-1} \quad \text{and} \quad T_k \equiv \left( I + \frac{1}{2} \tilde{D}_k \right) \left( I - \frac{1}{2} \tilde{D}_k \right)^{-1}. \]

Overall we have:

**Algorithm I: (The Intercept Method)**

1. Given $c^0$, compute the singular values $\{\sigma_i(c^0)\}_{i=1}^n$, the normalized left singular vectors $\{p_i(c^0)\}_{i=1}^m$, and the normalized right singular vectors $\{q_i(c^0)\}_{i=1}^n$ of $A(c^0)$. Let
   \[ P_0 = [p_0^1, \ldots, p_0^m] = [p_1(c^0), \ldots, p_m(c^0)] \in O(m), \]
   \[ Q_0 = [q_0^1, \ldots, q_0^n] = [q_1(c^0), \ldots, q_n(c^0)] \in O(n), \]
   \[ \sigma^0 = (\sigma_1(c^0), \ldots, \sigma_n(c^0))^T. \]

2. For $k = 0, 1, 2, \ldots$, until convergence, do:
   
   (a) Form the approximate Jacobian matrix $J_k$ by (7) and $a^k$ by (8).
   (b) Solve $c^{k+1}$ from the approximate Jacobian equation (6).
   (c) Form the matrix $A(c^{k+1})$ by (1).
   (d) Form the matrix $X_k \equiv P_k^T A(c^{k+1}) Q_k$.
   (e) Compute the skew-symmetric matrices $\tilde{C}_k$ and $\tilde{D}_k$ by (5) and (9)–(11).
   (f) Compute $P_{k+1} = [p_1^{k+1}, \ldots, p_m^{k+1}]$ and $Q_{k+1} = [q_1^{k+1}, \ldots, q_n^{k+1}]$ by solving
   \[ (I + \frac{1}{2} \tilde{C}_k) P_{k+1}^T = (I - \frac{1}{2} \tilde{C}_k) P_k^T, \]  \hspace{1cm} (12)
   \[ (I + \frac{1}{2} \tilde{D}_k) Q_{k+1}^T = (I - \frac{1}{2} \tilde{D}_k) Q_k^T. \]  \hspace{1cm} (13)

This approach is showed to converge at least quadratically in the root sense in [2]. For the definition of root-convergence rate, see Section 4 or [14, Chapter 9]. We point out that in each outer iteration (i.e. Step 2), we have to solve the linear equations (6) and (12)–(13). 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 (12)–(13) iteratively with only a few iterations. This is because that both $\|\tilde{C}_k\|$ and $\|\tilde{D}_k\|$ converge to zeros, see [2, equation (44)]. However, iterative methods may oversolve the approximate Jacobian equation (6) 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 last redundant inner iterations sharply is our goal in next section.

**3 The Inexact Intercept Method**

In this section, we propose an efficient inexact version of Algorithm I for the large problem. To decrease the computational cost, we use iterative methods to solve equations (6) and
(12)–(13). Especially, we solve equation (6) inexactly. That is, we will find an explicit stopping tolerance for (6), and then investigate the convergence analysis of the resulted procedure.

For general nonlinear equation \( f(c) = 0 \), the stopping criterion of inexact Newton methods is usually given in terms of \( f(c) \), see for instance [9, 10]. By (2), this will involve computing the exact singular values \( \{\sigma_i(c^k)\}_{i=1}^n \) of \( A(c^k) \) which are costly to compute. We will replace them by readily computational quantities as defined in (16) and (19) below. We will show in Section 4 that this replacement will retain superlinear convergence.

**Algorithm II: The Inexact Intercept Method**

1. Given \( c^0 \), compute the singular values \( \{\sigma_i(c^0)\}_{i=1}^n \), the orthogonal left singular vectors \( \{u_i(c^0)\}_{i=1}^m \) and right singular vectors \( \{v_i(c^0)\}_{i=1}^n \) of \( A(c^0) \). Let

\[
U_0 = [u_1^0, \ldots , u_m^0] = [u_1(c^0), \ldots , u_m(c^0)], \\
V_0 = [v_1^0, \ldots , v_n^0] = [v_1(c^0), \ldots , v_n(c^0)], \\
\sigma^0 = (\sigma_1(c^0), \sigma_2(c^0), \ldots , \sigma_n(c^0))^T.
\]

2. For \( k = 0, 1, 2, \ldots \), until convergence, do

(a) Form the approximate Jacobian matrix \( J_k \) and the vector \( a^k \):

\[
[J_k]_{ij} = (u_i^k)^T A_j v_i^k, \quad 1 \leq i, j \leq n. 
\]

\[
a_i^k = (u_i^k)^T A_0 v_i^k, \quad 1 \leq i \leq n. 
\]

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

\[
J_k c^{k+1} = \sigma^* - a^k + r^k,
\]

until the residual \( r^k \) satisfies

\[
\|r^k\| \leq \frac{\|\sigma^k - \sigma^*\|}{\|\sigma^*\|^\beta}, \quad \beta \in (1, 2].
\]

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

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

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

\[
[H_k]_{ij} = 0 \quad \text{for} \ n + 1 \leq i \neq j \leq m, \\
[H_k]_{ij} = \frac{-[H_k]_{ji}}{\sigma_j^*}, \quad \text{for} \ n + 1 \leq i \leq m, 1 \leq j \leq n, \\
[H_k]_{ij} = \frac{-[H_k]_{ji}}{\sigma_i^* (\sigma_j^*)^2 - (\sigma_i^*)^2}, \quad \text{for} \ 1 \leq i < j \leq n, \\
[K_k]_{ij} = \frac{-[K_k]_{ji}}{\sigma_i^* (\sigma_j^*)^2 - (\sigma_i^*)^2}, \quad \text{for} \ 1 \leq i < j \leq n.
\]
(f) Compute $U_{k+1} = [u_1^{k+1}, \ldots, u_m^{k+1}]$ and $V_{k+1} = [v_1^{k+1}, \ldots, v_n^{k+1}]$ by solving

\[
(I + \frac{1}{2}H_k)U_{k+1}^T = (I - \frac{1}{2}H_k)U_k^T, \quad (I + \frac{1}{2}K_k)V_{k+1}^T = (I - \frac{1}{2}K_k)V_k^T. 
\]

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

\[
\sigma_i^{k+1} = (u_i^{k+1})^T A(c^{k+1})v_i^{k+1}, \quad \text{for } 1 \leq i \leq n.
\]

We note that, since $U_0$ and $V_0$ are both orthogonal, and $H_k$ and $K_k$ are all skew-symmetric, we observe that the matrices $U_k$ and $V_k$ generated by the Cayley transforms in (17)–(18) should be orthogonal, i.e.

\[
U_k^TU_k = I \quad \text{and} \quad V_k^TV_k = I, \quad k = 0, 1, \ldots.
\]

To guarantee the orthogonality of $U_k$ and $V_k$, equations (17)–(18) 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 $c^0$ is close to $c^*$ sufficiently (see (73)). Then the matrices on the left-hand sides of (17) and (18) approach to the identity matrices in the limits. Thus, we can expect to solve (17) and (18) precisely by iterative methods with just a few iterations.

In algorithm II, the solution of (15) will be the costly step. In the next section, we will establish that the convergence rate of Algorithm II is equal to $\beta$ given in (16).

4 Convergence Analysis

In what follows, we let $c^k$ be the $k$th iterate produced by Algorithm II. Let the singular value decomposition of $A(c^*)$ be given by $A(c^*) = U(c^*)\Sigma(c^*)V(c^*)^T$ with $U(c^*) \in SO(n)$ and $V(c^*) \in SO(n)$. Let the Jacobian matrix $J(c^*)$ be defined by

\[
[J(c^*)]_{ij} = u_i(c^*)^T A_j v_j(c^*), \quad 1 \leq i, j \leq n,
\]

where $u_i(c^*)$ and $v_i(c^*)$ denote the $i$th columns of $U(c^*)$ and $V(c^*)$, respectively. As in [2], in what follows, we always assume that the Jacobian matrix $J(c^*)$ is nonsingular. By the continuity of the matrix inverse (see for instance [18, p.249]), there exist positive numbers $\xi$ and $c$ such that if there exist normalized vectors $u_i \in \mathbb{R}^m$ and $v_i \in \mathbb{R}^n$ satisfying

\[
\max\{|u_1, \ldots, u_n| - |u_1(c^*), \ldots, u_n(c^*)|, ||v_1, \ldots, v_n| - V(c^*)||\} \leq \xi,
\]

then the matrix $J = [u_i^T A_j v_i]$ is nonsingular and

\[
\|J^{-1}\| \leq c.
\]

Finally, we partition $\Sigma$, $U(c^*)$ and $U_k$ as

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

where $\Sigma_{*1} \in \mathbb{R}^{m \times n}$, $U_{11}(c^*) \in \mathbb{R}^{m \times n}$, and $U^{(k)}_{11} \in \mathbb{R}^{m \times n}$.
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 1** [12, Corollary 8.6.2] If $B$ and $B + E$ are in $\mathbb{R}^{m \times n}$ with $m \geq n$, then, for any $1 \leq k \leq n$,

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

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

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

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

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

**Lemma 3** [2, Lemma 5] If $E \in \mathbb{R}^{n \times n}$ and $\|E\| < 1$, then $I - \frac{1}{2}E$ is nonsingular and

$$\|(I + \frac{1}{2}E)(I - \frac{1}{2}E)^{-1} - (I + E)\| \leq \|E\|^2.$$
\textbf{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 (17)–(18), we have 
\[ U_k = U_{k-1}\Phi_{k-1} \quad \text{and} \quad V_k = V_{k-1}\Psi_{k-1}. \tag{26} \]
By Lemma 3, if \( \|H_{k-1}\| \leq 1 \) and \( \|V_{k-1}\| \leq 1 \), then we can write 
\[ \Phi_{k-1} = I + H_{k-1} + F_{k-1} \quad \text{and} \quad \Psi_{k-1} = I + K_{k-1} + G_{k-1}, \tag{27} \]
where 
\[ \|F_{k-1}\| \leq \|H_{k-1}\|^2 \quad \text{and} \quad \|G_{k-1}\| \leq \|K_{k-1}\|^2. \tag{28} \]
Notice from (21) that 
\[ U_k^T A(c^k)V_{k-1} = \Sigma_s + H_{k-1}\Sigma_s - \Sigma_sK_{k-1} + R_{k-1}. \]
Then by (26)–(27), a simple calculation gives rise to 
\[ U_k^T A(c^k)V_k = \Sigma_s + R_{k-1} + \Omega_{k-1}, \tag{29} \]
where 
\[ \Omega_{k-1} = F_{k-1}^T (\Sigma_s + H_{k-1}\Sigma_s - \Sigma_sK_{k-1} + R_{k-1}) (I + K_{k-1} + G_{k-1}) 
+ (I - H_{k-1}) (\Sigma_s + H_{k-1}\Sigma_s - \Sigma_sK_{k-1} + R_{k-1}) G_{k-1} 
+ (H_{k-1}\Sigma_s - \Sigma_sK_{k-1} + R_{k-1}) K_{k-1} - H_{k-1} (H_{k-1}\Sigma_s - \Sigma_sK_{k-1} + R_{k-1}) 
- H_{k-1} (\Sigma_s + H_{k-1}\Sigma_s - \Sigma_sK_{k-1} + R_{k-1}) K_{k-1}. \]
Thus if \( \|H_{k-1}\|, \|V_{k-1}\|, \) and \( \|\sigma^{k-1} - \sigma^*\| \) are small enough, then by (28) and (16), we get 
\[ \|\Omega_{k-1}\| = O(\|\sigma^{k-1} - \sigma^*\|2^\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2). \tag{30} \]
Equating the diagonal elements of (29) leads to 
\[ \sigma^k = \sigma^* + r^{k-1} + O(\|\sigma^{k-1} - \sigma^*\|2^\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2), \]
and by (16), we obtain 
\[ \|\sigma^k - \sigma^*\| = O(\|\sigma^{k-1} - \sigma^*\|^\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2), \tag{31} \]
On the other hand, combining (29) with (21) yields 
\[ U_k^T (A(c^{k+1}) - A(c^k))V_k = H_k\Sigma_s - \Sigma_sK_k + R_k - R_{k-1} - \Omega_{k-1}. \tag{32} \]
Now, based on (32), we will verify (22)– (25). To show (22), we note that the diagonal equations of (32) give rise to 
\[ J_k(c^{k+1} - c^k) = r^k - r^{k-1} + O(\|\sigma^{k-1} - \sigma^*\|^2\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2). \]
Thus 
\[ \|c^{k+1} - c^k\| = O(\|r^k\| + \|r^{k-1}\| + O(\|\sigma^{k-1} - \sigma^*\|^2\beta + \|H_{k-1}\|^2 + \|K_{k-1}\|^2). \tag{33} \]
By (16), (31), and (33), we obtain (22).
To get (23), let
\[ Z \equiv U_k^T (A(c^{k+1}) - A(c^k)) V_k - R_k + R_{k-1} + \Omega_{k-1} = \begin{bmatrix} Z_{11} \\ Z_{21} \end{bmatrix}^{n \times m-n}. \]
Noting that \( H_k \) has the form
\[ H_k = \begin{bmatrix} H_{11}^{(k)} & -H_{21}^{(k)} \\ H_{21}^{(k)} & 0 \end{bmatrix}, \quad H_{11}^{(k)} \in \mathbb{R}^{n \times n}. \]
Then, by (32), we obtain
\[ H_{11}^{(k)} \Sigma_{\ast 1} - \Sigma_{\ast 1} K_k = Z_{11}, \quad (34) \]
\[ H_{21}^{(k)} \Sigma_{\ast 1} = Z_{21}. \quad (35) \]
By Lemma 4, it follows from (34) that
\[ \| H_{11}^{(k)} \| = O(\| Z_{11} \|) = O(\| Z \|), \quad (36) \]
\[ \| K_k \| = O(\| Z_{11} \|) = O(\| Z \|). \quad (37) \]
Next, by (35), we have
\[ \| H_{21}^{(k)} \| = O(\| Z_{21} \|) = O(\| Z \|). \]
This, together with (36), yields
\[ \| H_k \| = O(\| H_{11}^{(k)} \| + 2\| H_{21}^{(k)} \|) = O(\| Z \|), \quad (38) \]
By Lemma 2, it follows from (16), (22), and (30)–(31) that
\[ \| Z \| = O(\| c^{k+1} - c^k \| + \| r^k \| + \| r^{k-1} \| + \| \Omega_{k-1} \|) \]
\[ = O(\| \sigma_{k-1} - \sigma^\ast \| \beta + \| H_{k-1} \|^2 + \| K_{k-1} \|^2). \quad (39) \]
By (31) and (37)–(39), we get (23).
Now we prove (24)–(25). By lemma 3, it follows from (26) and (23) that
\[ \| U_{k+1} - U_k \| = \| U_k (\Phi_k - I) \| = O(\| H_k \|) = O(\| \sigma_{k-1} - \sigma^\ast \| \beta + \| H_{k-1} \|^2 + \| K_{k-1} \|^2), \]
\[ \| V_{k+1} - V_k \| = \| V_k (\Psi_k - I) \| = O(\| K_k \|) = O(\| \sigma_{k-1} - \sigma^\ast \| \beta + \| H_{k-1} \|^2 + \| K_{k-1} \|^2). \]

By the above results, we can estimate the errors in \( \{ u_i(c^k) \}_{i=1}^n \) and \( \{ v_i(c^k) \}_{i=1}^n \) in terms of \( \| c^k - c^\ast \| \).

**Lemma 6 [2, Lemma 4]** Let the given singular values \( \{ \sigma_i^\ast \}_{i=1}^n \) be positive and distinct. Let the vectors \( u_i(c^k) \) and \( v_i(c^k) \) stand for the unit left and unit right singular vectors of \( A(c^k) \) respectively. Then there exist positive numbers \( \epsilon_4 \) and \( \gamma \) such that, if \( \| c^k - c^\ast \| \leq \epsilon_4 \), we have
\[ \| [u_1(c^k), \ldots, u_n(c^k)] - U_{11}(c^\ast) \| \leq \gamma \| c^k - c^\ast \|, \]
\[ \| [v_1(c^k), \ldots, v_n(c^k)] - V(c^\ast) \| \leq \gamma \| c^k - c^\ast \|. \]
4.2 R-Convergence Rate of Algorithm II

In the following, we show that the root-convergence rate of our method is at least $\beta$. Here, we recall the definition of root-convergence, see [14, Chap. 9].

**Definition 1** Let $\{x^k\}$ be a sequence with limit $x^*$. Then the numbers

\[
R_p(x^k) = \begin{cases} \limsup_{k \to \infty} \|x^k - x^*\|^{1/k}, & \text{if } p = 1, \\ \limsup_{k \to \infty} \|x^k - x^*\|^{1/p}, & \text{if } p > 1, \end{cases}
\]

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

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

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

We first prove that our method is locally convergent.

**Theorem 1** Let the given singular values $\{\sigma_i^k\}_{i=1}^n$ be positive and distinct and the Jacobian matrix $J(c^k)$ defined by (20) be nonsingular. Then there exists $\epsilon > 0$ such that if $\|c^0 - c^*\| \leq \epsilon$, the sequence $\{c^k\}$ generated by Algorithm II converges.

**Proof:** Suppose that the matrix $J_k$ defined in (14) is nonsingular, $\|\sigma^{k-1} - \sigma^*\| \leq \epsilon_5$, $\|H_{k-1}\| \leq \epsilon_5$, and $\|K_{k-1}\| \leq \epsilon_5$, where $\epsilon_5 \equiv \min\{1, \epsilon_1, \epsilon_2, \epsilon_3\}$ with $\epsilon_1$, $\epsilon_2$, and $\epsilon_3$ given in Lemma 5. By Lemma 5, there exists a constant $\tau > 1$ such that for any $k \geq 1$,

\[
\|c^{k+1} - c^k\| \leq \tau(\|\sigma^{k-1} - \sigma^*\| + \|H_{k-1}\|^2 + \|K_{k-1}\|^2),
\]

\[
\sqrt{\|\sigma^k - \sigma^*\|^2 + \|H_k\|^2 + \|K_k\|^2} \leq \tau(\|\sigma^{k-1} - \sigma^*\| + \|H_{k-1}\|^2 + \|K_{k-1}\|^2),
\]

\[
\|U_{k+1} - U_k\| \leq \tau(\|\sigma^{k-1} - \sigma^*\| + \|H_{k-1}\|^2 + \|K_{k-1}\|^2),
\]

\[
\|V_{k+1} - V_k\| \leq \tau(\|\sigma^{k-1} - \sigma^*\| + \|H_{k-1}\|^2 + \|K_{k-1}\|^2).
\]

We note that if

\[
\max\{\|U_{11}^{(k)} - U_{11}(c^*)\|, \|V_k - V(c^*)\|\} \leq \xi,
\]

then the matrix $J_k$ defined in (14) is nonsingular.

Let

\[
\phi = \max \left\{ 3\tau, \sqrt{n} \alpha, \left(1 + \frac{1}{\|\sigma^*\|^2}\right) \sqrt{n} \alpha (\alpha + 1) \left(\frac{2n\alpha^2}{d_\ast} + \frac{2}{\alpha^2}\right) \right\} > 1,
\]

where $d_\ast = \min_{i \neq j} (\sigma_i^k)^2 - (\sigma_j^k)^2$. By (42)–(45), it is derived that

\[
\max\{\|c^{k+1} - c^k\|, \|\sigma^{k-1} - \sigma^*\|, \|H_k\|, \|K_k\|, \|U_{k+1} - U_k\|, \|V_{k+1} - V_k\|\} \leq \phi \max\{\|\sigma^{k-1} - \sigma^*\|^\beta, \|H_{k-1}\|^\beta, \|K_{k-1}\|^\beta\}, \quad \text{for } k = 1, 2, \ldots.
\]

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

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

then the numbers $\{x^k\}$ defined in (14) are the root-convergence factors of $\{x^k\}$. The quantity

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

is called the root-convergence rate of $\{x^k\}$.\]
then for each \( k \geq 1 \), the following inequalities hold:

\[
\max\{\|U_{11}^{(k)} - U_{11}(c^*)\|, \|V_k - V(c^*)\|\} \leq \xi, \quad (48)
\]

\[
\max\{\|c^{k+1} - c^k\|, \|\sigma^k - \sigma^*\|, \|H_k\|, \|K_k\|, \|U_{k+1} - U_k\|, \|V_{k+1} - V_k\|\} \leq \phi^{1+\beta+\cdots+\beta^k}\|c_0 - c^*\|^{\beta^k},
\]

\[
\max\{\|c^{k+1} - c^k\|, \|\sigma^k - \sigma^*\|, \|H_k\|, \|K_k\|, \|U_{k+1} - U_k\|, \|V_{k+1} - V_k\|\} \leq \epsilon. \quad (50)
\]

We first estimate \( \|\sigma_0^0 - \sigma^*\|, \|H_0\|, \) and \( \|K_0\| \) in terms of \( \|c_0^0 - c^*\| \). By Lemmas 1 and 2, we have

\[
\max_i |\sigma_i(c_0^0) - \sigma_i^*| \leq \|A(c_0^0) - A(c^*)\| \leq \alpha \|c_0^0 - c^*\|.
\]

Thus

\[
\|\sigma_0^0 - \sigma^*\| \leq \sqrt{n} \max_i |\sigma_i(c_0^0) - \sigma_i^*| \leq \sqrt{n} \alpha \|c_0^0 - c^*\| \leq \phi \|c_0^0 - c^*\| \leq \epsilon_5.
\]

Let \( \Sigma_0 \equiv \text{diag}(\sigma_0(c_0^0), \ldots, \sigma_n(c_0^0)) \in \mathbb{R}^{m \times n} \). It is easily to know that

\[
U_0^T A(c_0^0) V_0 = \Sigma_0.
\]

Notice from (21) that

\[
U_0^T A(c) V_0 = \Sigma_s + H_0 \Sigma_s - \Sigma_s K_0 + R_0.
\]

Taking the difference between (52) and (53) yields

\[
U_0^T (A(c^1) - A(c^0)) V_0 = \Sigma_s - \Sigma_0 + H_0 \Sigma_s - \Sigma_s K_0 + R_0.
\]

The diagonal equations of (54) give rise to

\[
J_0(c^1 - c^0) = \sigma^* - \sigma_0^0 + r^0.
\]

By Lemma 6 and using (47), we have

\[
\max\{\|U_{11}^{(0)} - U_{11}(c^*)\|, \|V_0 - V(c^*)\|\} \leq \gamma \|c^0 - c^*\| \leq \frac{\xi}{2},
\]

and then \( J_0 \) is nonsingular and \( \|J_0^{-1}\| \leq c \). Therefore, by (16), (51), and (55), we have

\[
\|c^1 - c^0\| \leq c(\|\sigma^* - \sigma_0^0\| + \|r_0\|) \leq (1 + \frac{1}{\|\sigma^*\|}) \sqrt{n} \alpha \|c_0^0 - c^*\|.
\]

To estimate \( \|H_0\| \) and \( \|K_0\| \) in terms of \( \|c_0^0 - c^*\| \), let

\[
Z \equiv U_0^T (A(c^1) - A(c^0)) V_0 - \Sigma_s + \Sigma_0 - R_0 = \begin{bmatrix} Z_{11} & \ldots & Z_{1n} \\ Z_{21} & \ldots & Z_{2n} \end{bmatrix}^{m-n}. \]

Then, by Lemma 2 and using (16), (51), and (57), we have

\[
\|Z\| \leq \|A(c^1) - A(c^0)\| + \|\sigma^* - \sigma_0^0\| + \|r_0\| \leq \alpha \|c^1 - c^0\| + (1 + \frac{1}{\|\sigma^*\|}) \|\sigma^* - \sigma_0^0\|
\]

\[
\leq (1 + \frac{1}{\|\sigma^*\|}) \sqrt{n} \alpha (c_0 + 1) \|c_0^0 - c^*\|. \quad (58)
\]

11
Noting 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 (54), we obtain

$$H_{11}^{(0)} \Sigma_{s1} - \Sigma_{s1} K_0 = Z_{11}, \quad (59)$$

and

$$H_{21}^{(0)} \Sigma_{s1} = Z_{21}, \quad (60)$$

By Lemma 4, it follows from (59) that

$$\|H_{11}^{(0)}\| \leq \frac{2n \sigma_1^*}{d_s} \|Z_{11}\|, \quad (61)$$

$$\|K_0\| \leq \frac{2n \sigma_1^*}{d_s} \|Z_{11}\| \leq \frac{2n \sigma_1^*}{d_s} \|Z\|. \quad (62)$$

On the other hand, by (60), we have

$$\|H_{21}^{(0)}\| \leq \frac{1}{\sigma_n^*} \|Z_{21}\|. \quad (63)$$

This, together with (61), yields

$$\|H_0\| \leq \|H_{11}^{(0)}\| + 2\|H_{21}^{(0)}\| \leq \frac{2n \sigma_1^*}{d_s} \|Z_{11}\| + \frac{2}{\sigma_n^*} \|Z_{21}\| \leq \left( \frac{2n \sigma_1^*}{d_s} + \frac{2}{\sigma_n^*} \right) \|Z\|. \quad (64)$$

By (58) and (62)–(63), we get

$$\|H_0\| \leq \left( 1 + \frac{1}{\|\sigma^*\|_2} \right) \sqrt{n} \alpha (ca + 1) \left( \frac{2n \sigma_1^*}{d_s} + \frac{2}{\sigma_n^*} \right) \|e_0 - c^*\| \leq \phi \|e_0 - c^*\| \leq \epsilon_5, \quad (65)$$

By Lemma 3, we have from (26) that

$$\|U_1 - U_0\| \leq \|\Phi_0 - I\| = 2\|H_0\| \leq 2 \phi \|e_0 - c^*\| \leq \frac{\xi}{4}, \quad (66)$$

$$\|V_1 - V_0\| \leq \|\Psi_0 - I\| \leq 2\|K_0\| \leq 2 \phi \|e_0 - c^*\| \leq \frac{\xi}{4}. \quad (67)$$

Now, we verify that (48) holds for $k = 1$. From (56) and (66)–(67),

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

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

Next, we show that (49) holds for $k = 1$. By (46), (51), and (64)–(65),

$$\max\{\|c^2 - c^1\|, \|\sigma^1 - \sigma^*\|, \|H_1\|, \|K_1\|, \|U_2 - U_1\|, \|V_2 - V_1\|\} \leq \phi \max\{\|\sigma^0 - \sigma^*\|, \|H_0\|, \|K_0\|\} \leq \phi^{1+\beta} \|e_0 - c^*\|^\beta. \quad (68)$$
If we let $\varphi \equiv \phi^{\frac{m}{\epsilon}} \epsilon$, then by (47), we have

$$\varphi^\beta \leq \epsilon < 1.$$  \hspace{1cm} (69)

Thus by (68),

$$\max\{\|c^2 - c^1\|, \|\sigma - \sigma^*\|, \|H_k\|, \|K_k\|, \|U_{k+1} - U_k\|, \|V_{k+1} - V_k\|\}$$

$$\leq \phi^{1+\beta} \|c^0 - \sigma^*\| \leq (\phi \frac{1+\beta}{\beta}) \|c^0 - \sigma^*\| \leq \varphi^\beta \leq \epsilon.$$  \hspace{1cm} (70)

That is, (50) holds for $k = 1$.

Now, we assume that (48)–(50) hold for all positive integer less than or equal to $k - 1$. We first prove that (48) holds for $k$. In fact, by (49), we have, for $j = 1, 2, \ldots, k - 1$,

$$\max\{\|U_{j+1} - U_j\|, \|V_{j+1} - V_j\|\} \leq \phi^{1+\beta+\cdots+\beta} \|c^0 - \sigma^*\| \leq \left(\phi \frac{1+\beta}{\beta}\right)^{\beta}$$

$$= \left(\phi^{1+\beta+\cdots+\beta} \|c^0 - \sigma^*\|\right)^{\beta} \leq (\varphi^\beta \|c^0 - \sigma^*\|) \leq \varphi^\beta.$$  \hspace{1cm} (71)

Then, by (70), (67), (56), and (69), we get

$$\|V_k - V(\sigma^*)\| \leq \sum_{j=1}^{k-1} \|V_{j+1} - V_j\| + \|V_1 - V_0\| + \|V_0 - V(\sigma^*)\|$$

$$\leq \sum_{j=1}^{k-1} \varphi^{\beta} + \frac{\xi}{4} + \frac{\xi}{4} \leq \sum_{j=1}^{k-1} \varphi^{1+\beta j} + \frac{\xi}{4} + \frac{\xi}{4}$$

$$\leq \sum_{j=1}^{k-1} (\varphi^{\beta j}) + \frac{\xi}{4} + \frac{\xi}{4} \leq \varphi^{\beta \ln \beta} + \frac{\xi}{4} + \frac{\xi}{4}$$

$$\leq \frac{\xi}{1 - \varphi^{\beta \ln \beta}} + \frac{\xi}{4} + \frac{\xi}{4} \leq \frac{\xi}{4} + \frac{\xi}{4} + \frac{\xi}{4} + \frac{\xi}{4} = \xi.$$  \hspace{1cm} (72)

By the same argument, we can prove that $\|U_{11}^{(k)} - U_{11}(\sigma^*)\| \leq \xi$. Therefore, (48) holds for $k$.

To prove that (49) holds for $k$, we use (46):

$$\max\{\|c^{k+1} - c^k\|, \|\sigma^k - \sigma^*\|, \|H_k\|, \|K_k\|, \|U_{k+1} - U_k\|, \|V_{k+1} - V_k\|\}$$

$$\leq \phi \max\{\|\sigma^{k-1} - \sigma^*\|, \|H_{k-1}\|, \|K_{k-1}\|\}$$

$$\leq \phi (\varphi^{1+\beta + \cdots + \beta^{k-1}})^{\beta}$$

$$= \phi^{1+\beta + \cdots + \beta^k} \|c^0 - \sigma^*\|^{\beta^k}.$$  \hspace{1cm} (72)
To verify that (50) holds for \( k \). By (72), we have
\[
\max \{ \| c^{k+1} - c^k \|, \| \sigma^k - \sigma^* \|, \| H_k \|, \| K_k \|, \| U_{k+1} - U_k \|, \| V_{k+1} - V_k \| \} \\
\leq \left( \phi^{\frac{1+\beta+\ldots+eta^k}{eta}} \| c^0 - c^* \| \right)^{\beta^k} \\
\leq (\phi^{\frac{\beta}{\beta^k}} \| c^0 - c^* \|)^{\beta^k} \leq \varphi^{\beta^k} \\
\leq \varphi^\beta \leq \epsilon. \tag{73}
\]

Thus we have proved that (48)–(50) hold for any \( k \geq 1 \).

Finally, we show the local convergence of the sequence \( \{ c^k \} \). From (73), we have, for any integer \( m \geq 1 \),
\[
\| c^{k+m} - c^k \| \leq \sum_{l=1}^{m} \| c^{k+l} - c^{k+l-1} \| \leq \sum_{l=1}^{m} \varphi^{\beta^{k+l-1}} = \sum_{l=1}^{m} \left( \varphi^{\beta^{k-1}} \right)^{\beta^{l}} \leq \sum_{l=1}^{m} (\varphi^{\beta^{k-1}})^{1+l\ln\beta} \\
\leq \sum_{l=1}^{m} \left( \varphi^{\beta^{k-1}} \ln\beta \right)^l \leq \frac{\varphi^{\beta^{k-1}} \ln\beta - (\varphi^{\beta^{k-1}} \ln\beta)^{m+1}}{1 - \varphi^{\beta^{k-1}} \ln\beta}. \tag{74}
\]

This shows that \( \{ c^k \} \) is a Cauchy sequence since \( \varphi \leq \frac{1}{\beta} < 1 \). Therefore, there exists a vector \( c^+ \in \mathbb{R}^n \) such that the sequence \( \{ c^k \} \) converges to \( c^+ \).

\[ \square \]

**Remark.** We remark that \( c^+ \) may not equal to the original solution \( c^* \), see for instance [2].

We end this section by establishing the root convergence rate of our method.

**Theorem 2** Under the same conditions as in Theorem 1, the sequence \( \{ c^k \} \) converges to the limit \( c^+ \) with root-convergence rate at least equal to \( \beta \).

**Proof:** By Theorem 1, we know that \( \{ c^k \} \) is locally convergent with
\[
\lim_{k \to \infty} c^k = c^+.
\]

Since \( \varphi < 1 \), let \( m \to \infty \) in (74), we have, for each \( k \geq 1 \),
\[
\| c^k - c^+ \| \leq \frac{\varphi^{\beta^{k-1}} \ln\beta}{1 - \varphi^{\beta^{k-1}} \ln\beta} \leq \omega (\varphi \ln\beta)^{\beta^{k-1}},
\]
where \( \omega = \frac{1}{1 - \varphi \ln\beta} > 1 \).

We now estimate the root-convergence factors of \( \{ c^k \} \) defined in (40) for different values of \( p \):

1. If \( p = 1 \), then
   \[
   R_1 \{ c^k \} = \limsup_{k \to \infty} \| c^k - c^+ \|^{1/k} \leq \limsup_{k \to \infty} \omega^{1/k} (\varphi \ln\beta)^{\beta^{k-1}/k} = 0.
   \]

2. If \( 1 < p < \beta \), then
   \[
   R_p \{ c^k \} = \limsup_{k \to \infty} \| c^k - c^+ \|^{1/p^k} \leq \limsup_{k \to \infty} \omega^{1/p^k} (\varphi \ln\beta)^{(\beta/p)^k} = 0.
   \]
3. If \( p = \beta \), then
\[
R_p \{c^k\} = \limsup_{k \to \infty} \|c^k - c^+\|^{1/\beta^k} \leq \limsup_{k \to \infty} \omega^{1/p^k} \varphi^{\ln \beta / \beta} = \varphi^{\ln \beta / \beta} < 1.
\]

4. If \( p > \beta \), then
\[
R_p \{c^k\} = \limsup_{k \to \infty} \|c^k - c^+\|^{1/p^k} \leq \limsup_{k \to \infty} \omega^{1/p^k} (\varphi^{\ln \beta / \beta})^{(\beta/p)^k} = 1.
\]

Therefore, \( R_p \{c^k\} = 0 \) for any \( p \in [1, \beta) \) and \( R_p \{c^k\} \leq 1 \) for any \( p \in [\beta, \infty) \). Thus according to (41), \( \Omega_R(c^+) \geq \beta \).

\[\Box\]

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 \texttt{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 \( c^* \in \mathbb{R}^n \) and compute the singular values \( \{\sigma_i^*\}_{i=1}^n \) of \( A(c^*) \) as the prescribed singular values. Because of the local convergence of both algorithms, the initial guess \( c^0 \) was formed by chopping the components of \( c^* \) 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 (6), (12)–(13) of algorithm I, and (15), (17)–(18) of Algorithm II were all solved by the QMR method [15] via the Matlab-provided \texttt{QMR} function. To guarantee the orthogonality of the iterates \( P_k \) and \( Q_k \) and \( U_k \) and \( V_k \), equations (12)–(13) and (17)–(18) are solved iteratively up to the machine precision (\( \approx 2.22 \times 10^{-16} \)). The iterative method may converge slow. To speed up the convergence, one may use some preconditioner for equations (6) and (15). 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 [8, 13]. 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 \( c^k \), the iterant at the \( k \)th Newton iteration, as the initial guess for solving the approximate Jacobian equations (6) and (15) iteratively in the \( (k+1) \)th outer iteration. In Algorithm II, the stopping tolerance for (15) is given in (16). Since Algorithm I is the exact version of Algorithm II, (6) should be solved iteratively up to the machine precision. Here, we try to use a large value \( 10^{-14} \) as the stopping tolerance for (6), and compare the two algorithms. When the problem size is larger, the stopping tolerance for (6) 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
\[
\|P_k^T A(c^k)Q_k - \Sigma_s\|_F \leq 10^{-10} \quad \text{and} \quad \|U_k^T A(c^k)V_k - \Sigma_s\|_F \leq 10^{-10}.
\]
Table 1: Averaged total numbers of outer and inner iterations.

<table>
<thead>
<tr>
<th>n</th>
<th>Alg. I</th>
<th>Alg. II</th>
<th>β in Alg. II</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1.1</td>
<td>1.2</td>
</tr>
<tr>
<td>a)</td>
<td></td>
<td>(N_o)</td>
<td>3.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(N_i)</td>
<td>411</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(N_o)</td>
<td>3.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(N_i)</td>
<td>25.5</td>
</tr>
<tr>
<td>b)</td>
<td></td>
<td>(N_o)</td>
<td>3.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(N_i)</td>
<td>966</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(N_o)</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(N_i)</td>
<td>30.8</td>
</tr>
<tr>
<td>c)</td>
<td></td>
<td>(N_o)</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(N_i)</td>
<td>43.4</td>
</tr>
</tbody>
</table>

We now report our experimental results. Table 1 lists the total numbers of outer iterations \(N_o\) averaged over ten tests and 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 analysis of the two algorithms. However, under small \(N_o\), 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.

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 Euclidean vector norm error \(e\) between the current approximation and its limits point \(c^+\) (not necessarily the same as \(c^*\)). Figure 1 depicts 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 (12)–(13) and (17)–(18), 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.

Finally, we remark that we also retried the same tests with different iterative solvers (e.g. BiCG [17] and CGS [16]) 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 ((6) and (15)) are nonsymmetric and dense, it needs further study to find a better preconditioner, see for instance [3] 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.

<table>
<thead>
<tr>
<th>Outer iteration</th>
<th>a)</th>
<th>b)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1st</td>
<td>2nd</td>
</tr>
<tr>
<td>Alg. I</td>
<td>12.0</td>
<td>7.5</td>
</tr>
<tr>
<td>Alg. II with $\beta = 2$</td>
<td>14.0</td>
<td>8.0</td>
</tr>
<tr>
<td>Alg. II with $\beta = 1.5$</td>
<td>12.5</td>
<td>7.7</td>
</tr>
</tbody>
</table>

Acknowledgments

The author would like to thank Prof. Raymond Chan and Prof. Shu-fang Xu for their valuable discussions and suggestions when we began this paper. This work was partially supported by the HKRGC Grant CUHK 400503.

References


