TAIWANESE JOURNAL OF MATHEMATICS
Vol. 16, No. 1, pp. 367-386, February 2012
This paper is available online at http://tjm.math.ntu.edu.tw

AN ULM-LIKE CAYLEY TRANSFORM METHOD FOR INVERSE EIGENVALUE PROBLEMS

Weiping Shen* and Chong Li

Abstract. We propose an Ulm-like Cayley transform method for solving inverse eigenvalue problems, which avoids solving approximate Jacobian equations comparing with other known methods. A convergence analysis of this method is provided and the R-quadratic convergence property is proved under the assumption of the distinction of the given eigenvalues. Numerical experiments are given in the last section and comparisons with the inexact Cayley transform method [1] are made.

1. Introduction

Inverse eigenvalue problems (IEPs) arise in a variety of applications such as inverse Sturm-Liouville's problem, inverse vibrating string problem, nuclear spectroscopy and molecular spectroscopy (see [2, 3, 7, 9, 14, 15, 18, 20, 23, 25, 26, 27, 30, 32]). In particular, a recent survey paper on structured inverse eigenvalue problems by Chu and Golub (see [7]) is a good reference for these applications. In many of these applications, the problem size n can be large. For example, large inverse Toeplitz eigenvalue problems and large discrete inverse Sturm-Liouville's problems considered in [7, 27].

The IEPs we considered here is defined as follows. Let $\mathbf{c}= (c_1, c_2, \ldots, c_n)^T \in \mathbb{R}^n$ and $\{A_i\}_{i=0}^n$ be a sequence of real symmetric $n \times n$ matrices. Define

$$A(\mathbf{c}) := A_0 + \sum_{i=1}^{n} c_i A_i \tag{1.1}$$

Received September 8, 2010, accepted July 2, 2011.
Communicated by Jen-Chih Yao.
2010 *Mathematics Subject Classification*: 65F18, 65F10, 65F15.
Key words and phrases: Nonlinear equation, Inverse eigenvalue problem, Moser's method, Inexact Cayley transform method.
The first author was supported in part by the National Natural Science Foundation of China (Grant 111013797) and by Zhejiang Provincial Natural Science Foundation of China (Grant Y6100375); the second author was supported in part by the National Natural Science Foundation of China (Grant 11171300) and by Zhejiang Provincial Natural Science Foundation of China (Grant Y6110006).
*Corresponding author

and denote its eigenvalues by $\{\lambda_i(\mathbf{c})\}_{i=1}^n$ with the ordering $\lambda_1(\mathbf{c}) \le \lambda_2(\mathbf{c}) \le \cdots \le \lambda_n(\mathbf{c})$. Let $\{\lambda_i^*\}_{i=1}^n$ be given with $\lambda_1^* \le \lambda_2^* \le \cdots \le \lambda_n^*$. Then the IEP considered here is to find a vector $\mathbf{c}^* \in \mathbb{R}^n$ such that

$$\lambda_i(\mathbf{c}^*) = \lambda_i^* \quad \text{for each } i = 1, 2, \ldots, n. \tag{1.2}$$

The vector $\mathbf{c}^*$ is called a solution of the IEP (1.2).

Recall that solving the IEP (1.2) is equivalent to solving the equation $\mathbf{f}(\mathbf{c}) = \mathbf{0}$ on $\mathbb{R}^n$, where the function $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$ is defined by

$$\mathbf{f}(\mathbf{c}) = (\lambda_1(\mathbf{c}) - \lambda_1^*, \lambda_2(\mathbf{c}) - \lambda_2^*, \ldots, \lambda_n(\mathbf{c}) - \lambda_n^*)^T \quad \text{for any } \mathbf{c} \in \mathbb{R}^n.$$

Based on this equivalence, Newton's method can be applied to the IEP, and it converges quadratically (see [12, 21, 34]). However, Newton's method has two disadvantages from the point of view of practical calculation: one is that it requires solving a complete eigenproblem for the matrix $A(\mathbf{c})$, and the other that it requires solving exactly the Jacobian equations. Since then many researchers committed themselves to overcoming these two disadvantages of Newton's method. To overcome the first disadvantage, different Newton-like methods, where each outer iteration, instead of the exact eigenvectors of the matrix $A(\mathbf{c})$, adopts approximations to them, have been proposed and studied in [5, 6, 12, 34]. In particular, Friedland et al. considered in [12] a type of Newton-like method where the approximate eigenvectors were found by using the one-step inverse power method, and the Cayley transform method which forms approximate Jacobian equations by applying matrix exponentials and Cayley transforms. To overcome the second disadvantage and alleviate the over-solving problem, inexact methods for solving the IEPs have been proposed. For example, the inexact Newton-like method and the inexact Cayley transform method was proposed in [4] and [1] respectively. Motivated by Moser's method and Ulm's method (see [10, 13, 17, 19, 24, 28, 29, 33, 35]), we have proposed in paper [31] an Ulm-like method for solving the IEPs (with $A_0 = 0$) , which avoids solving the approximate Jacobian equations and hence is more stable comparing with the inexact Newton-like method. A Numerical example for which the Ulm-like method converges but not the inexact Newton-like method is provided in that paper.

Combining Ulm's method with the Cayley transform method, we propose an Ulm-like Cayley transform method for solving the IEPs in this paper, which also avoids solving the Jacobian equations in each outer iteration. It should be noted that the Ulm-like Cayley transform method covers both the cases $A_0 = 0$ and $A_0 \neq 0$. Under the classical assumption (which is also used in [1, 4]) that the given eigenvalues are distinct and the Jacobian matrix $J(\mathbf{c}^*)$ is nonsingular, we prove that this method converges with R-quadratic convergence. Comparing with the inexact Cayley transform method in [1], the Ulm-like Cayley transform method seems more stable, and reduces the difficulty though they have the same costs. Numerical experiments are given in the last section to illustrate the comparisons with the inexact Cayley transform method.

2. Ulm-like Cayley Transform Method

Let $[1, n]$ denote the set of $\{1, 2, \ldots, n\}$. As usual, let $\mathbb{R}^{n\times n}$ denote the set of all real $n \times n$ matrices. Let $\|\cdot\|$ denote the 2-norm in $\mathbb{R}^n$. The induced 2-norm in $\mathbb{R}^{n\times n}$ is also denoted by $\|\cdot\|$, i.e.,

$$\|A\| := \sup_{\mathbf{x}\in\mathbb{R}^n,\, \mathbf{x}\neq\mathbf{0}} \frac{\|A\mathbf{x}\|}{\|\mathbf{x}\|} \quad \text{for each } A \in \mathbb{R}^{n\times n}.$$

Let $\|\cdot\|_F$ denote the Frobenius norm in $\mathbb{R}^{n\times n}$. Then

$$\|A\| \leq \|A\|_F \quad \text{for each } A \in \mathbb{R}^{n\times n}.$$

Let $\mathbf{c}= (c_1, c_2, \ldots, c_n)^T \in \mathbb{R}^n$ and $\{A_i\}_{i=1}^n \subset \mathbb{R}^{n\times n}$ be symmetric. As in (1.1), define

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

Let $\lambda_1(\mathbf{c}) \leq \lambda_2(\mathbf{c}) \leq \cdots \leq \lambda_n(\mathbf{c})$ be the eigenvalues of the matrix $A(\mathbf{c})$, and let $\{\mathbf{q}_i(\mathbf{c})\}_{i=1}^n$ be the normalized eigenvectors corresponding to $\{\lambda_i(\mathbf{c})\}_{i=1}^n$. Define $J(\mathbf{c}) = ([J(\mathbf{c})]_{ij})$ by

$$[J(\mathbf{c})]_{ij} := \mathbf{q}_i(\mathbf{c})^T A_j \mathbf{q}_i(\mathbf{c}) \quad \text{for any } i, j \in [1, n].$$

Let $\{\lambda_i^*\}_{i=1}^n$ be given with $\lambda_1^* \leq \lambda_2^* \leq \cdots \leq \lambda_n^*$ and write $\boldsymbol{\lambda}^* = (\lambda_1^*, \lambda_2^*, \ldots, \lambda_n^*)^T$. Let $\mathbf{c}^*$ be the solution of the IEP, i.e.,

$$\lambda_i(\mathbf{c}^*) = \lambda_i^* \quad \text{for each } i \in [1, n].$$

As shown in [1, 4, 12], in the case when the given eigenvalues $\{\lambda_i^*\}_{i=1}^n$ are distinct, the eigenvalues of $A(\mathbf{c})$ are distinct too for any point $\mathbf{c}$ in some neighborhood of $\mathbf{c}^*$. It follows that the function $\mathbf{f}(\cdot)$ is analytic in the same neighborhood, and $J(\mathbf{c})$ is the Jacobian matrix of $\mathbf{f}$ at $\mathbf{c}$ in this neighborhood (see. [1, 4]). Recall that Newton's method, which converges quadratically (see. [21, 34]), involves solving a complete eigenproblem for the matrix $A(\mathbf{c})$. However, if we only compute it approximately, we may still have fast convergence. This results in the following Cayley transform method which forms approximate Jacobian equations by applying matrix exponentials and Cayley transforms.

Algorithm 1. Cayley transform method

1. Given $\mathbf{c}^0$, compute the orthonormal eigenvectors $\{\mathbf{q}_i(\mathbf{c}^0)\}_{i=1}^n$ of $A(\mathbf{c}^0)$. Let $P_0 = [\mathbf{p}_1^0, \ldots, \mathbf{p}_n^0] = [\mathbf{q}_1(\mathbf{c}^0), \ldots, \mathbf{q}_n(\mathbf{c}^0)]$.

2. For $k = 0, 1, \ldots$ until convergence, do:
   (a) Form the approximate Jacobian matrix J_k and the vector $\mathbf{b}^k$ as follows:
   $$[J_k]_{ij} = (\mathbf{p}_i^k)^T A_j \mathbf{p}_i^k \quad \text{for each } i, j \in [1, n],$$
   $$[\mathbf{b}^k]_i = (\mathbf{p}_i^k)^T A_0 \mathbf{p}_i^k \quad \text{for each } i \in [1, n]. \tag{2.1}$$
   (b) Solve $\mathbf{c}^{k+1}$ from the approximate Jacobian equation
   $$J_k \mathbf{c}^{k+1} = \boldsymbol{\lambda}^* - \mathbf{b}^k. \tag{2.2}$$
   (c) Form the skew-symmetric matrix Y_k:
   $$[Y_k]_{ij} = \frac{(\mathbf{p}_i^k)^T A(\mathbf{c}^{k+1}) \mathbf{p}_j^k}{\lambda_j^* - \lambda_i^*} \quad \text{for each } i, j \in [1, n] \quad and \quad i \neq j. \tag{2.3}$$
   (d) Compute $P_{k+1} = [\mathbf{p}_1^{k+1}, \ldots, \mathbf{p}_n^{k+1}] = [\mathbf{v}_1^{k+1}, \ldots, \mathbf{v}_n^{k+1}]^T$ by solving
   $$(I + \tfrac{1}{2} Y_k) \mathbf{v}_j^{k+1} = \mathbf{h}_j^k \quad \text{for each } j \in [1, n], \tag{2.4}$$
   where $\mathbf{h}_j^k$ is the jth column of $H_k = (I - \frac{1}{2} Y_k) P_k^T$.

In [12], it was proved that the Cayley transform method **Algorithm 1** converges with R-quadratic convergence. Note that in **Algorithm 1**, systems (2.2) and (2.4) are solved exactly. Usually, one solves these systems by iterative methods, in particular in the case when n is large. One could expect that it requires only a few iterations to solve (2.4) iteratively. This is due to the fact that, as $\{\mathbf{c}^k\}$ converges to $\mathbf{c}^*$, $\|Y_k\|$ converges to zero (cf. [1, 12]). Consequently, the coefficient matrix on the left-hand side of (2.4) approaches the identity matrix in the limit, and therefore (2.4) can be solved efficiently by iterative methods. On the other hand, as for the approximate Jacobian equation (2.2), iterative methods may bring an over-solving problem in the sense that the last few iterations before convergence are usually insignificant as far as the convergence of the outer iteration is concerned. This over-solving of the inner iterations will cause a waste of time and does not improve the efficiency of the whole method. To alleviate the over-solving problem and improve the efficiency in solving the IEP, system (2.2) is solved in [1] approximately rather than exactly, and the following inexact Cayley transform method was proposed there.

Algorithm 2. Inexact Cayley transform method

1. Given $\mathbf{c}^0$, compute the orthonormal eigenvectors $\{\mathbf{q}_i(\mathbf{c}^0)\}_{i=1}^n$ and the eigenvalues $\{\lambda_i(\mathbf{c}^0)\}_{i=1}^n$ of $A(\mathbf{c}^0)$. Let $P_0 = [\mathbf{p}_1^0, \ldots, \mathbf{p}_n^0] = [\mathbf{q}_1(\mathbf{c}^0), \ldots, \mathbf{q}_n(\mathbf{c}^0)]$ and $\boldsymbol{\rho}^0 = (\lambda_1(\mathbf{c}^0), \ldots, \lambda_n(\mathbf{c}^0))^T$.

2. For $k = 0, 1, \ldots$ until convergence, do:
   (a) Same as (a) in **Algorithm 1**
   (b) Solve $\mathbf{c}^{k+1}$ inexactly from the approximate Jacobian equation

$$J_k \mathbf{c}^{k+1} = \boldsymbol{\lambda}^* - \mathbf{b}^k + \mathbf{r}^k, \tag{2.5}$$

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

$$\|\mathbf{r}^k\| \leq \frac{\|\boldsymbol{\rho}^k - \boldsymbol{\lambda}^*\|^\beta}{\|\boldsymbol{\lambda}^*\|^\beta}, \quad \beta \in (1, 2].$$

   (c) Same as (c) in **Algorithm 1**
   (d) Same as (d) in **Algorithm 1**
   (e) Compute $\boldsymbol{\rho}^{k+1} = (\rho_1^{k+1}, \ldots, \rho_n^{k+1})^T$ by

$$\rho_i^{k+1} = (\mathbf{p}_i^{k+1})^T A(\mathbf{c}^{k+1}) \mathbf{p}_i^{k+1} \quad \text{for each } i = 1, 2, \ldots, n.$$

Under the assumption that the given eigenvalues $\{\lambda_i^*\}_{i=1}^n$ are distinct and the Jacobian matrix $J(\mathbf{c}^*)$ is invertible, it was proved in [1] that the inexact Cayley transform method converges locally with root-convergence rate equal to β. In each outer iteration of the inexact Cayley transform method, an approximate Jacobian equation is required to solve. This still can be costly sometimes especially when $\mathbf{c}^k$ is close to the solution $\mathbf{c}^*$. Furthermore, solving (2.5) may involve some preconditioning problem.

Moser's method (see [17, 24, 28]) to solve operator equations in Banach spaces is defined as follows. Let X, Y be (real or complex) Banach spaces, and let $D \subseteq X$ be an open subset. Consider the general operator equation:

$$f(x) = 0, \tag{2.6}$$

where $f : D \subseteq X \longrightarrow Y$ is a nonlinear operator with continuous Fréchet derivative f'. Given $x_0 \in D$ and $B_0 \in \mathcal{L}(Y, X)$, Moser's method to find solutions of equation (2.6) is defined as follows:

$$\begin{cases} x_{k+1} = x_k - B_k f(x_k) \\ B_{k+1} = 2B_k - B_k f'(x_k) B_k \end{cases} \quad \text{for each } k = 0, 1, \ldots.$$

The convergence rate of Moser's method is $(1 + \sqrt{5})/2 = 1.61\cdots$ (see [24]). However, quadratic convergence rate can be obtained when the sequence $\{B_k\}$ generated by

$$B_{k+1} = 2B_k - B_k f'(x_{k+1}) B_k \quad \text{for each } k = 0, 1, \ldots.$$

This is Ulm's method introduced in [33] and has been further studied in [10, 13, 17, 19, 28, 29, 35]. R-quadratic convergence of Ulm's method was established in [10, 19, 35] under the classical assumption that the derivative f' is Lipschitz continuous around the solution. Compared with Newton's method, the advantage of Moser's method and Ulm's method is that Jacobian equations are not required to solve in each step. Motivated by Moser's method and Ulm's method, we propose the following Ulm-like Cayley transform method for solving the IEP, which also avoids solving the approximate Jacobian equations in each step.

Algorithm 3. Ulm-like Cayley transform method

1. Given $\mathbf{c}^0 \in \mathbb{R}^n$ and $B_0 \in \mathbb{R}^{n\times n}$ be such that

$$\|I - B_0 J(\mathbf{c}^0)\| \le \mu, \tag{2.7}$$

   where μ is a positive constant. Compute the orthonormal eigenvectors $\{\mathbf{q}_i(\mathbf{c}^0)\}_{i=1}^n$ of $A(\mathbf{c}^0)$. Let $P_0 = [\mathbf{p}_1^0, \ldots, \mathbf{p}_n^0] = [\mathbf{q}_1(\mathbf{c}^0), \ldots, \mathbf{q}_n(\mathbf{c}^0)]$ and $J_0 = J(\mathbf{c}^0)$. Compute the vector $\mathbf{b}^0$ by (2.1).

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

   (a) Compute $\mathbf{c}^{k+1}$ by

$$\mathbf{c}^{k+1} = \mathbf{c}^k - B_k(J_k\mathbf{c}^k - \boldsymbol{\lambda}^* + \mathbf{b}^k). \tag{2.8}$$

   (b) Same as (c) in **Algorithm 1**.

   (c) Same as (d) in **Algorithm 1**.

   (d) Form the approximate Jacobian matrix J_{k+1} and the vector $\mathbf{b}^{k+1}$ as follows:

$$[J_{k+1}]_{ij} = (\mathbf{p}_i^{k+1})^T A_j \mathbf{p}_i^{k+1} \quad \text{for each } i, j \in [1, n],$$

$$[\mathbf{b}^{k+1}]_i = (\mathbf{p}_i^{k+1})^T A_0 \mathbf{p}_i^{k+1} \quad \text{for each } i \in [1, n].$$

   (e) Compute the matrix B_{k+1} by

$$B_{k+1} = 2B_k - B_k J_{k+1} B_k.$$

Remark 2.1. Note that (2.4) implies

$$P_{k+1} = P_k(I + \tfrac{1}{2}Y_k)(I - \tfrac{1}{2}Y_k)^{-1} \quad \text{for each } k = 0, 1, \ldots. \tag{2.9}$$

Since P_0 is an orthogonal matrix and $\{Y_k\}$ are skew-symmetric matrices, we see that the matrices $\{P_k\}$ generated by (2.4) must be orthogonal, i.e.,

$$P_k^T P_k = P_k P_k^T = I \quad \text{for each } k = 0, 1, \ldots.$$

To maintain the orthogonality of P_k, that would mean that (2.4) cannot be solved inexactly. However, we will see in Section 3 that $\|Y_k\|$ converges to zero (see (3.7), (3.8), (3.10) and (3.15)). Consequently, the matrix on the left-hand side of (2.4) approaches the identity matrix in the limit. Therefore we can expect to solve (2.4) accurately by iterative methods using just a few iterations.

Remark 2.2. The main difference of the Ulm-like Cayley transform method and the inexact Cayley transform method is that the step of solving the approximate Jacobian equation (2.2) in the inexact Cayley transform method is replaced by computing the product of matrices, the operation cost of which is still $O(n^3)$, the same as that of solving the Jacobian equation. However, computing the product of matrices is simpler than solving equations. Therefore, the Ulm-like Cayley transform method reduces significantly the difficulty of the problem. In particular, the parallel computation techniques can be applied in the Ulm-like Cayley transform method to improve the computational efficiency.

3. Convergence Analysis

In this section, we carry on a convergence analysis of the Ulm-like Cayley transform method. Let $\{\lambda_i^*\}_{i=1}^n$ be given with $\lambda_1^* \le \lambda_2^* \le \cdots \le \lambda_n^*$. Let $\mathbf{c}^*$ be the solution of the IEP and let $Q_* = [q_1(\mathbf{c}^*), \ldots, q_n(\mathbf{c}^*)]$ be the orthogonal matrix of the eigenvectors of $A(\mathbf{c}^*)$. Then the matrix Q_* satisfies

$$Q_*^T A(\mathbf{c}^*) Q_* = \wedge_*, \tag{3.1}$$

where $\wedge_* = \mathrm{diag}(\lambda_1^*, \ldots, \lambda_n^*)$. As the standard assumption in [1, 4], we assume that the given eigenvalues $\{\lambda_i^*\}_{i=1}^n$ are distinct and that the Jacobian matrix $J(\mathbf{c}^*)$ is nonsingular. Let $\mathbf{c}^k$ be the kth iteration of the method, $\{\lambda_i(\mathbf{c}^k)\}_{i=1}^n$ and $\{\mathbf{q}_i(\mathbf{c}^k)\}_{i=1}^n$ be the eigenvalues and normalized eigenvectors of $A(\mathbf{c}^k)$ respectively, i.e.,

$$A(\mathbf{c}^k)\mathbf{q}_i(\mathbf{c}^k) = \lambda_i(\mathbf{c}^k)\mathbf{q}_i(\mathbf{c}^k) \quad \text{and} \quad \mathbf{q}_i(\mathbf{c}^k)^T \mathbf{q}_j(\mathbf{c}^k) = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{if } i \ne j. \end{cases}$$

Below, we prove that if B_0 approximates $J(\mathbf{c}^0)^{-1}$ and the initial guess $\mathbf{c}^0$ is closed to the solution $\mathbf{c}^*$, then the sequence $\{\mathbf{c}^k\}$ generated by the Ulm-like Cayley transform method converges locally to $\mathbf{c}^*$ with R-quadratic convergence. For this purpose, we need the following four lemmas. The first and the third lemmas have been presented in [4] and [12, Corollary 3.1] respectively; while the proof of the second one is similar to that of [22, Lemma 3.2].

Lemma 3.1. *Suppose that $\{\lambda_i^*\}_{i=1}^n$ are distinct. Then there exist positive numbers δ_0 and ρ_0 such that the following assertion holds for each $\mathbf{c} \in \mathbf{B}(\mathbf{c}^*, \delta_0)$.*

$$\| \mathbf{q}_i(\mathbf{c}) - \mathbf{q}_i(\mathbf{c}^*) \| \le \rho_0 \| \mathbf{c} - \mathbf{c}^* \| \quad \textit{for each } i \in [1, n].$$

Lemma 3.2. *Let $\{\boldsymbol{\omega}_i\}_{i=1}^n \subset \mathbb{R}^n$ be unit vectors approximating $\mathbf{q}_i(\mathbf{c}^*)$. Let $J_{\boldsymbol{\omega}}$ be the matrix defined by $[J_{\boldsymbol{\omega}}]_{ij} = (\boldsymbol{\omega}_i)^T A_j \boldsymbol{\omega}_i$ for each $i, j \in [1, n]$ and let $\mathbf{b}_{\boldsymbol{\omega}}$ be the vector defined by $[\mathbf{b}_{\boldsymbol{\omega}}]_i = (\boldsymbol{\omega}_i)^T A_0 \boldsymbol{\omega}_i$ for each $i \in [1, n]$.Then*

$$\| J_{\boldsymbol{\omega}} \mathbf{c}^* - \boldsymbol{\lambda}^* + \mathbf{b}_{\boldsymbol{\omega}} \| \leq 2n \cdot \max_i |\lambda_i^*| \cdot \max_i \| \boldsymbol{\omega}_i - \mathbf{q}_i(\mathbf{c}^*) \|^2 .$$

Lemma 3.3. *There exist two positive numbers δ_1 and ρ_1 such that, for any orthogonal matrix P with $\|P - Q_*\| < \delta_1$, the skew-symmetric matrix X defined by $e^X = P^T Q_*$ satisfies*

$$\|X\| \leq \rho_1 \|P - Q_*\|.$$

Let $\{P_k\}$ be defined by (2.9). Define

$$E_k := P_k - Q_* \quad \text{for each } k = 0, 1, \ldots. \tag{3.2}$$

Then we have the following key lemma.

Lemma 3.4. *Suppose that the given eigenvalues $\{\lambda_i^*\}_{i=1}^n$ are distinct and the Jacobian matrix $J(\mathbf{c}^*)$is invertible. Then there exist positive numbers δ_2 and ρ_2 such that, for any $k = 0, 1, \ldots$, if $\|\mathbf{c}^{k+1} - \mathbf{c}^*\| \leq \delta_2$ and $\|E_k\| \leq \delta_2$ then*

$$\|E_{k+1}\| \leq \rho_2 (\|\mathbf{c}^{k+1} - \mathbf{c}^*\| + \|E_k\|^2). \tag{3.3}$$

Proof. Let $k = 1, 2, \ldots$ and consider the matrix X_k defined by $e^{X_k} = P_k^T Q_*$. Since P_k is orthogonal, it follows from (3.1) that

$$e^{X_k} \wedge_* e^{-X_k} = P_k^T A(\mathbf{c}^*) P_k. \tag{3.4}$$

Note that

$$e^X = I + X + O(\|X\|^2) \quad \text{for each matrix } X. \tag{3.5}$$

Combining this and (3.4), we have that

$$\wedge_* + X_k \wedge_* - \wedge_* X_k = P_k^T A(\mathbf{c}^*) P_k + O(\|X_k\|^2),$$

Let $i, j = 1, \ldots, n$ with $i \neq j$. Then the above equality implies that

$$[X_k]_{ij} = \frac{1}{\lambda_j^* - \lambda_i^*} (\mathbf{p}_i^k)^T A(\mathbf{c}^*) \mathbf{p}_j^k + O(\|X_k\|^2).$$

By the definition of Y_k in (2.3), one has that

$$[X_k]_{ij} - [Y_k]_{ij} = \frac{1}{\lambda_j^* - \lambda_i^*} (\mathbf{p}_i^k)^T (A(\mathbf{c}^*) - A(\mathbf{c}^{k+1})) \mathbf{p}_j^k + O(\|X_k\|^2).$$

As $A(\cdot)$ is Lipschitz continuous, it follows that

$$|[X_k]_{ij} - [Y_k]_{ij}| = O(\|\mathbf{c}^{k+1} - c^*\| + \|X_k\|^2).$$

Consequently,

$$\|X_k - Y_k\| \leq \|X_k - Y_k\|_F = O(\|\mathbf{c}^{k+1} - \mathbf{c}^*\| + \|X_k\|^2) \tag{3.6}$$

and so $\|Y_k\| = O(\|\mathbf{c}^{k+1} - \mathbf{c}^*\| + \|X_k\|)$. Let $C > 0$ be such that

$$\|Y_k\| \leq C(\|\mathbf{c}^{k+1} - \mathbf{c}^*\| + \|X_k\|). \tag{3.7}$$

Take $0 < \delta_2 < \min\left\{\delta_1, \frac{1}{(1+\rho_1)C}\right\}$, where δ_1 and ρ_1 are the positive numbers determined by Lemma 3.3. Assume that $\|\mathbf{c}^{k+1} - \mathbf{c}^*\| < \delta_2$ and $\|E_k\| < \delta_2$. Then $\|E_k\| < \delta_1$ and Lemma 3.3 is applicable to getting

$$\|X_k\| \leq \rho_1\|E_k\|. \tag{3.8}$$

Thus (3.7) entails that

$$\|Y_k\| \leq C(\|\mathbf{c}^{k+1} - \mathbf{c}^*\| + \rho_1\|E_k\|) \leq C(1 + \rho_1)\delta_2 \leq 1.$$

Consequently

$$\left\|\left(I - \tfrac{1}{2}Y_k\right)^{-1}\right\| \leq \frac{1}{1 - \frac{1}{2}\|Y_k\|} \leq 2. \tag{3.9}$$

Below we will show that there exists $\rho_2 > 0$ such that (3.3) holds. Granting this, the proof is complete. To this end, we note by (3.2) and (2.9) that

$$E_{k+1} = P_k[(I+\tfrac{1}{2}Y_k)(I-\tfrac{1}{2}Y_k)^{-1}-e^{X_k}] = P_k[(I+\tfrac{1}{2}Y_k)-e^{X_k}(I-\tfrac{1}{2}Y_k)](I-\tfrac{1}{2}Y_k)^{-1}.$$

Applying (3.5), one has

$$\begin{aligned} E_{k+1} &= P_k[(I + \tfrac{1}{2}Y_k) - (I + X_k + O(\|X_k\|^2))(I - \tfrac{1}{2}Y_k)](I - \tfrac{1}{2}Y_k)^{-1} \\ &= P_k[Y_k - X_k + \tfrac{1}{2}X_kY_k + O(\|X_k\|^2)](I - \tfrac{1}{2}Y_k)^{-1}. \end{aligned}$$

Since P_k is orthogonal, it follows that

$$\|E_{k+1}\| \leq \left[\|Y_k - X_k\| + O(\|X_k\| \cdot \|Y_k\| + \|X_k\|^2)\right] \|(I - \tfrac{1}{2}Y_k)^{-1}\|.$$

Thus (3.3) is seen to hold by (3.6)-(3.9). ■

Now we present the main result of this paper which shows that the Ulm-like Cayley transform method converges with R-quadratic convergence.

Theorem 3.1. *Suppose that* $\{\lambda_i^*\}_{i=1}^n$ *are distinct and that the Jacobian matrix* $J(\mathbf{c}^*)$ *is invertible. Then there exist* $\tau \in (0,1]$, $\delta \in (0,\tau)$ *and* $\mu \in [0,\delta]$ *such that, for each* $\mathbf{c}^0 \in \mathbf{B}(\mathbf{c}^*,\delta)$ *and* $B_0 \in \mathbb{R}^{n\times n}$ *satisfying* (2.7), *the sequence* $\{\mathbf{c}^k\}$ *generated by* **Algorithm 3** *with initial point* $\mathbf{c}^0$ *converges to* $\mathbf{c}^*$. *Moreover, the following estimates hold for each* $k = 0, 1, \ldots$.

$$\|\mathbf{c}^k - \mathbf{c}^*\| \le \tau \left(\frac{\delta}{\tau}\right)^{2^k}, \tag{3.10}$$

$$\|I - B_k J_k\| \le \tau \left(\frac{\delta}{\tau}\right)^{2^k}. \tag{3.11}$$

Proof. Let $\delta_0, \rho_0 \in (0,+\infty)$ be the constants determined by Lemma 3.1 such that $H_1\|J(\mathbf{c}^*)^{-1}\|\delta_0 < 1$ where

$$H_1 := 2n^2\rho_0 \cdot \max_j \|A_j\|.$$

Let $\delta_2 \in (0,\delta_1]$ and $\rho_2 \in (0,+\infty)$ be the constants determined by Lemma 3.4. Moreover, we write for simplicity,

$$\bar\rho = \frac{\|J(\mathbf{c}^*)^{-1}\|}{1 - H_1\|J(\mathbf{c}^*)^{-1}\|\delta_0} \quad \text{and} \quad H_2 = 4n^2\rho_0^2\bar\rho \cdot \max_i |\lambda_i^*|.$$

Set

$$\tau = \min\left\{\frac{1}{1+H_2},\ \frac{\sqrt{n}\rho_0}{\rho_2(1+H_2+n\rho_0^2)},\ \frac{1}{(1+4\bar\rho H_1)^2}\right\}. \tag{3.12}$$

Clearly

$$\tau \le 1. \tag{3.13}$$

Take δ and μ such that

$$0 < \delta < \min\left\{\delta_0,\ \delta_2,\ \tau,\ \frac{\delta_2}{\sqrt{n}\rho_0},\ \frac{1}{\bar\rho H_1}\right\} \quad \text{and} \quad 0 \le \mu \le \delta.$$

Then, one has by the definition of $\bar\rho$ that

$$\delta < \frac{1}{\bar\rho H_1} = \frac{1 - H_1\|J(\mathbf{c}^*)^{-1}\|\delta_0}{H_1\|J(\mathbf{c}^*)^{-1}\|} < \frac{1}{H_1\|J(\mathbf{c}^*)^{-1}\|}.$$

Thus for any matrix A,

$$\|A - J(\mathbf{c}^*)\| \le H_1\delta \Longrightarrow A^{-1} \text{ exists and } \|A^{-1}\| \le \bar\rho. \tag{3.14}$$

Below we shall show that τ, δ and μ are as desired. To do this, let $k = 0, 1, \ldots$ and consider the following conditions:

(3.15) $$\|E_k\| \le \sqrt{n}\rho_0\tau\left(\frac{\delta}{\tau}\right)^{2^k};$$

(3.16) $$\|J_k - J(\mathbf{c}^*)\| \le H_1\tau\left(\frac{\delta}{\tau}\right)^{2^k} \quad \text{and} \quad \|J_k^{-1}\| \le \bar{\rho};$$

(3.17) $$\|B_k\| \le 2\bar{\rho} \quad \text{and} \quad \|\mathbf{c}^{k+1} - \mathbf{c}^*\| \le (1+H_2)\tau^2\left(\frac{\delta}{\tau}\right)^{2^{k+1}}.$$

Then we have the following implications:

(3.18) $$(3.15) \Longrightarrow (3.16)$$

and

(3.19) $$[(3.15) + (3.10) + (3.11)] \Longrightarrow (3.17).$$

To prove the first implication, we suppose that (3.15) holds. Note that

(3.20) $$\|\mathbf{p}_i^k - \mathbf{q}_i(\mathbf{c}^*)\| \le \|P_k - Q_*\| = \|E_k\| \le \sqrt{n}\rho_0\tau\left(\frac{\delta}{\tau}\right)^{2^k} \quad \text{for each } i \in [1,n].$$

Then, for any $i,\, j \in [1,n]$,

$$\begin{aligned} |\,[J_k]_{ij} - [J(\mathbf{c}^*)]_{ij}\,| &= |\,[\mathbf{p}_i^k - \mathbf{q}_i(\mathbf{c}^*)]^T A_j \mathbf{p}_i^k - \mathbf{q}_i(\mathbf{c}^*)^T A_j[\mathbf{q}_i(\mathbf{c}^*) - \mathbf{p}_i^k]\,| \\ &\le 2\,\|\,A_j\,\| \cdot \|\,\mathbf{p}_i^k - \mathbf{q}_i(\mathbf{c}^*)\,\| \\ &\le 2\sqrt{n}\rho_0\,\|\,A_j\|\tau\left(\frac{\delta}{\tau}\right)^{2^k}, \end{aligned}$$

which together with the definition of H_1 gives

$$\|J_k - J(\mathbf{c}^*)\| \le \|J_k - J(\mathbf{c}^*)\|_F \le 2n^{\frac{3}{2}}\rho_0 \cdot \max_j \|A_j\|\tau\left(\frac{\delta}{\tau}\right)^{2^k} \le H_1\tau\left(\frac{\delta}{\tau}\right)^{2^k}.$$

Thus, the first inequality of (3.16) is proved; while the second inequality follows from (3.14) because $\|J_k - J(\mathbf{c}^*)\| \le H_1\delta$ (noting that $\delta < \tau$). Therefore, the implication (3.18) is proved.

To verify the second implication, suppose that (3.15), (3.10) and (3.11) hold. Then (3.16) holds by implication (3.18) and so $\|J_k^{-1}\| \le \bar{\rho}$. Furthermore,

$$\|B_kJ_k\| \le 1 + \|I - B_kJ_k\| \le 1 + \tau\left(\frac{\delta}{\tau}\right)^{2^k}.$$

Therefore,

$$\|B_k\| \le \|B_kJ_k\| \cdot \|J_k^{-1}\| \le \left[1+\tau\left(\frac{\delta}{\tau}\right)^{2^k}\right]\bar{\rho} \le (1+\tau)\bar{\rho} \le 2\bar{\rho} \tag{3.21}$$

(noting that $\tau \le 1$ by (3.13)). As for the estimate of $\|\mathbf{c}^{k+1}-\mathbf{c}^*\|$, we note by (2.8) that

$$\begin{aligned} \mathbf{c}^{k+1}-\mathbf{c}^* &= \mathbf{c}^k-\mathbf{c}^*-B_kJ_k\mathbf{c}^k+B_k\boldsymbol{\lambda}^*-B_k\mathbf{b}^k \\ &= (I-B_kJ_k)(\mathbf{c}^k-\mathbf{c}^*)-B_k(J_k\mathbf{c}^*-\boldsymbol{\lambda}^*+\mathbf{b}^k). \end{aligned} \tag{3.22}$$

Since $\max_i \|\mathbf{p}_i^k-\mathbf{q}_i(\mathbf{c}^*)\| \le \sqrt{n}\rho_0\tau\left(\frac{\delta}{\tau}\right)^{2^k}$ by (3.20), it follows from Lemma 3.2 that

$$\begin{aligned} \|J_k\mathbf{c}^*-\boldsymbol{\lambda}^*+\mathbf{b}^k\| &\le 2n\cdot\max_i|\lambda_i^*|\cdot\max_i\|\mathbf{p}_i^k-\mathbf{q}_i(\mathbf{c}^*)\|^2 \\ &\le 2n^2\rho_0^2\cdot\max_i|\lambda_i^*|\tau^2\left(\frac{\delta}{\tau}\right)^{2^{k+1}}. \end{aligned} \tag{3.23}$$

By (3.10), (3.11) and using (3.21), (3.23), we conclude from (3.22) that

$$\begin{aligned} \|\mathbf{c}^{k+1}-\mathbf{c}^*\| &\le \|I-B_kJ_k\|\cdot\|\mathbf{c}^k-\mathbf{c}^*\|+\|B_k\|\cdot\|J_k\mathbf{c}^*-\boldsymbol{\lambda}^*+\mathbf{b}^k\| \\ &\le \tau^2\left(\frac{\delta}{\tau}\right)^{2^{k+1}}+4n^2\rho_0^2\bar{\rho}\cdot\max_i|\lambda_i^*|\tau^2\left(\frac{\delta}{\tau}\right)^{2^{k+1}} \\ &= (1+H_2)\tau^2\left(\frac{\delta}{\tau}\right)^{2^{k+1}}, \end{aligned}$$

where the equality holds because of the definition of H_2. This together with (3.21) completes the proof of the second implication.

Below we will show that (3.10), (3.11) and (3.15) hold for each $k \ge 0$. We will proceed by mathematical induction. Clearly, (3.10) and (3.11) for $k=0$ are trivial by assumptions (noting that $\mu \le \delta$). Moreover, by Lemma 3.1,

$$\|E_0\| \le \|E_0\|_F \le \sqrt{n}\cdot\max_i\|\mathbf{q}_i(\mathbf{c}^0)-\mathbf{q}_i(\mathbf{c}^*)\| \le \sqrt{n}\rho_0\|\mathbf{c}^0-\mathbf{c}^*\| \le \sqrt{n}\rho_0\delta.$$

Thus (3.15) is true for $k=0$.

Now assume that (3.10), (3.11) and (3.15) hold for all $k \le m-1$. Then, recalling that $\delta \le \tau$ and $\delta \le \frac{\delta_2}{\sqrt{n}\rho_0}$, one has that

$$\|E_{m-1}\| \le \sqrt{n}\rho_0\tau\left(\frac{\delta}{\tau}\right)^{2^{m-1}} \le \sqrt{n}\rho_0\delta \le \delta_2$$

Moreover, by implication (3.19) (with $k=m-1$), one has

$$\|B_{m-1}\| \le 2\bar{\rho}. \tag{3.24}$$

Hence

$$\|\mathbf{c}^m - \mathbf{c}^*\| \le (1+H_2)\tau^2 \left(\frac{\delta}{\tau}\right)^{2^m} < \delta < \delta_2,$$

Applying Lemma 3.4, we obtain

$$\begin{aligned}\|E_m\| &\le \rho_2(\|\mathbf{c}^m - \mathbf{c}^*\| + \|E_{m-1}\|^2)\\ &\le \rho_2(1+H_2+n\rho_0^2)\tau^2 \left(\frac{\delta}{\tau}\right)^{2^m} \le \sqrt{n}\rho_0\tau \left(\frac{\delta}{\tau}\right)^{2^m},\end{aligned}$$

where the last inequality holds because of (3.12). Thus (3.10) and (3.15) are seen to hold for $k = m$. To show that (3.11) holds for $k = m$, using implication (3.18) (with $k = m-1$, m), one gets that

$$\|J_{m-1} - J(\mathbf{c}^*)\| \le H_1\tau \left(\frac{\delta}{\tau}\right)^{2^{m-1}} \quad \text{and} \quad \|J_m - J(\mathbf{c}^*)\| \le H_1\tau \left(\frac{\delta}{\tau}\right)^{2^m}.$$

Hence

$$\|J_m - J_{m-1}\| \le \|J_m - J(\mathbf{c}^*)\| + \|J_{m-1} - J(\mathbf{c}^*)\| \le 2H_1\tau \left(\frac{\delta}{\tau}\right)^{2^{m-1}}. \tag{3.25}$$

Recalling that $B_k = 2B_{k-1} - B_{k-1}J_kB_{k-1}$ for each $k = 1, 2, \ldots$, we have

$$I - B_mJ_m = (I - B_{m-1}J_m)^2 = (I - B_{m-1}J_{m-1} - B_{m-1}(J_m - J_{m-1}))^2.$$

Then by (3.24), (3.25) and using the inductional assumption (3.11) (with $k = m-1$), we have that

$$\begin{aligned}\|I - B_mJ_m\| &\le (\|I - B_{m-1}J_{m-1}\| + \|B_{m-1}\| \cdot \|J_m - J_{m-1}\|)^2\\ &\le \left[\tau\left(\frac{\delta}{\tau}\right)^{2^{m-1}} + 4\bar{\rho}H_1\tau\left(\frac{\delta}{\tau}\right)^{2^{m-1}}\right]^2\\ &= (1+4\bar{\rho}H_1)^2\tau^2\left(\frac{\delta}{\tau}\right)^{2^m}.\end{aligned}$$

Note by (3.12) that $\tau(1+4\bar{\rho}H_1)^2 \le 1$. It follows that

$$\|I - B_mJ_m\| \le \tau\left(\frac{\delta}{\tau}\right)^{2^m},$$

that is (3.11) holds for $k = m$ and completes the proof. ∎

4. Numerical Experiments

In this section, we illustrate the convergence performance of the Ulm-like Cayley transform method on two examples. For comparison, the inexact Cayley transform method [1] is also tested. Our aim is to show that the outer iteration number required for convergence of the Ulm-like Cayley transform method is comparable to that of the inexact Cayley transform method [1] with large β. In the first example, we consider the inverse Toeplitz eigenvalue problem which was considered in [1, 4, 27, 32].

Example 4.1. We use Toeplitz matrices as our $\{A_i\}_{i=1}^n$ in (1.1):

$$A_1 = I,\ A_2 = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 1 & 0 & 1 & \ddots & \vdots \\ 0 & 1 & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & 0 & 1 \\ 0 & \cdots & 0 & 1 & 0 \end{pmatrix}, \ldots, A_n = \begin{pmatrix} 0 & 0 & \cdots & 0 & 1 \\ 0 & \ddots & \ddots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \ddots & \ddots & 0 \\ 1 & 0 & \cdots & 0 & 0 \end{pmatrix}.$$

Thus $A(\mathbf{c})$ is a symmetric Toeplitz matrix with the first column equal to $\mathbf{c}$.

All our tests were done in Matlab. In [27], very large inverse Toeplitz eigenvalue problem were solved on parallel architectures. Here we consider three problem sizes: $n = 100$, 200, and 300. For each value of n, we constructed ten n-by-n test problems. For each test problem, we first generate $\mathbf{c}^*$ randomly. Then we compute the eigenvalues $\{\lambda_i^*\}_{i=1}^n$ of $A(\mathbf{c}^*)$ as the prescribed eigenvalues. Since both algorithms are locally convergent, the initial guess $\mathbf{c}^0$ is formed by chopping the components of $\mathbf{c}^*$ to four decimal places for $n = 100$ and five decimal places for $n = 200,\ 300$. For both algorithms, the stopping tolerance for the outer (Newton) iterations is 10^{-10}.

Linear systems (2.4) and (2.5) are all solved iteratively by the QMR method (cf. [1, 4, 11]) using the Matlab-provided QMR function. At the $(k+1)$th iteration, we use $\mathbf{v}_j^k$ as the initial guess of the inverse power equation (2.4), and $\mathbf{c}^k$ as the initial guess of the approximate Jacobian equation (2.5). The stopping tolerance for the system (2.5) is given as in the equation. We also set the maximum number of iterations allowed to 400 for all inner iterations. To speed up the convergence, we use the Matlab-provided Modified ILU (MILU) preconditioner: LUINC(A, [drop-tolerance, 1, 1, 1]) which is one of the most versatile pre-conditioners for unstructured matrices [1, 8, 16]. The drop tolerance we use here is 0.05 for all the three problem sizes.

For $n = 100$, 200, and 300, the convergence performances of **Algorithms 2** and **3** are illustrated in Table 1, where "ite." represents the averaged total numbers of outer iterations on ten test problems. Note by (2.7) that B_0 is an approximation to $J(\mathbf{c}^0)^{-1}$. Here, we take $B_0 = J(\mathbf{c}^0)^{-1}$ for the Ulm-like Cayley transform method. Since the inexact Cayley transform method converges with convergence rate β, we present its convergence performances with large β. From this table, we can see that for $n = 100, 200$, and 300, the outer iteration numbers of the Ulm-like Cayley transform method are comparable to that of the inexact Cayley transform method. However, it should be noted that, by computing approximations to the inverse of Jacobian matrices, the Ulm-like Cayley transform method avoids solving the approximate Jacobian equation in each outer iteration. This will be very attractive when the approximate Jacobian equation (2.5) is difficult to solve. Moreover, when the size n is large, we can obtain the sequence $\{B_k\}$ by parallel computation which can further improve the computational efficiency.

To further illustrate the convergence performance of the Ulm-like Cayley transform method, Table 2 gives the averaged values of $\|\mathbf{c}^k - \mathbf{c}^*\|$ and the averaged total numbers of outer iterations of the Ulm-like Cayley transform method with different μ. "ite." is the same as in Table 1. From Table 2, we can see that, for the Ulm-like Cayley transform method, the convergence performance in the case of $B_0 \approx J(\mathbf{c}^0)^{-1}$ is comparable to that in the case of $B_0 = J(\mathbf{c}^0)^{-1}$.

Below we consider an example of non-Toeplitz matrices, where the matrices $\{J_k\}$ are large though both the inexact Cayley transform method and the Ulm-like Cayley transform method converge.

Example 4.2. Given $B = I + VV^T$, where

$$V = \begin{pmatrix} 1 & -1 & -3 & -5 & -6 \\ 1 & 1 & -2 & -5 & -17 \\ 1 & -1 & -1 & 5 & 18 \\ 1 & 1 & 1 & 2 & 0 \\ 1 & -1 & 2 & 0 & 1 \\ 1 & 1 & 3 & 0 & -1 \\ 2.5 & .2 & .3 & .5 & .6 \\ 2 & -.2 & .3 & .5 & .8 \end{pmatrix}_{8\times 5}$$

Define the matrices $\{A_k\}$ from B as follows:

$$A_k = b_{kk} e_k e_k^T + \sum_{j=1}^{k-1} b_{kj} (e_k e_j^T + e_j e_k^T) \quad \text{for each } k = 1, 2, \ldots, 8,$$

where e_k is the kth column of the identity matrix. Now suppose that

$$\mathbf{c}^* = (1.043890381645, 1.065644751834, 1.091344270553, 1.023155499528,$$
$$0.997448154933, 0.991139967277, 1.094291990723, 0.996548791312)^T.$$

Then,

Table 1: Convergence performances of **Algorithms 2** and **3**

n	k	**Algorithm 2**				**Algorithm 3**
		β=1.5	β=1.6	β=1.8	β=2	
100	0	$5.2699e-4$	$5.2699e-4$	$5.2699e-4$	$5.2699e-4$	$5.2699e-4$
	1	$4.4550e-6$	$4.4558e-6$	$4.4424e-6$	$4.4393e-6$	$4.4405e-6$
	2	$1.4358e-8$	$1.4377e-8$	$1.4389e-8$	$1.4386e-8$	$3.4099e-8$
	3	$4.2053e-13$	$4.1859e-13$	$4.2816e-13$	$4.3118e-13$	$1.9025e-11$
	4	0.0000	0.0000	0.0000	0.0000	0.0000
	ite.	3.0	3.0	3.0	3.0	3.0
200	0	$7.8370e-5$	$7.8370e-5$	$7.8370e-5$	$7.8370e-5$	$7.8370e-5$
	1	$3.2570e-7$	$3.2622e-7$	$3.2588e-7$	$3.2591e-7$	$3.2590e-7$
	2	$8.8251e-10$	$8.8469e-10$	$8.8385e-10$	$8.8373e-10$	$2.5966e-9$
	3	$1.5325e-14$	$1.6391e-14$	$1.4518e-14$	$1.6375e-14$	$3.6458e-13$
	4	0.0000	0.0000	0.0000	0.0000	0.0000
	ite.	3.0	3.0	3.0	3.0	3.0
300	0	$9.0310e-5$	$9.0310e-5$	$9.0310e-5$	$9.0310e-5$	$9.0310e-5$
	1	$4.6435e-7$	$4.5768e-7$	$4.5765e-7$	$4.5768e-7$	$4.5769e-7$
	2	$8.5937e-10$	$8.5906e-10$	$8.5818e-6$	$8.5827e-10$	$2.1156e-9$
	3	$3.6959e-14$	$3.7737e-14$	$3.8281e-14$	$3.6711e-14$	$2.5052e-12$
	4	0.0000	0.0000	0.0000	0.0000	0.0000
	ite.	3.0	3.0	3.0	3.0	3.0

Table 2: Convergence performances of **Algorithm 3** with different μ

n	k	$\mu=1e-1$	$\mu=1e-2$	$\mu=1e-3$	$\mu=1e-4$	$\mu=0$
100	0	$5.2699e-4$	$5.2699e-4$	$5.2699e-4$	$5.2699e-4$	$5.2699e-4$
	1	$5.2220e-5$	$6.3180e-6$	$4.3839e-6$	$4.4320e-6$	$4.4405e-6$
	2	$5.4890e-7$	$3.1382e-8$	$3.3495e-8$	$3.4035e-8$	$3.4099e-8$
	3	$1.4570e-10$	$1.330035e-11$	$1.8233e-11$	$1.8936e-11$	$1.9025e-11$
	4	$4.9600e-15$	0.0000	0.0000	0.0000	0.0000
	4	0.0000	0.0000	0.0000	0.0000	0.0000
	ite.	3.8	3.0	3.0	3.0	3.0
200	0	$7.8370e-5$	$7.8370e-5$	$7.8370e-5$	$7.8370e-5$	$7.8370e-5$
	1	$7.8225e-6$	$8.2955e-7$	$3.2999e-7$	$3.2542e-7$	$3.2590e-7$
	2	$7.8654e-8$	$2.2786e-9$	$2.5538e-9$	$2.5922e-9$	$2.5966e-9$
	3	$1.4053e-11$	$3.0477e-13$	$3.5299e-13$	$3.6297e-13$	$3.6458e-13$
	4	0.0000	0.0000	0.0000	0.0000	0.0000
	ite.	3.0	3.0	3.0	3.0	3.0
300	0	$9.0310e-5$	$9.0310e-5$	$9.0310e-5$	$9.0310e-5$	$9.0310e-5$
	1	$8.9605e-6$	$9.2954e-7$	$4.4864e-7$	$4.5598e-7$	$4.5769e-7$
	2	$8.9243e-8$	$2.2761e-9$	$2.1114e-9$	$2.1149e-9$	$2.1156e-9$
	3	$2.4742e-11$	$2.4436e-12$	$2.5055e-12$	$2.5053e-12$	$2.5052e-12$
	4	0.0000	0.0000	0.0000	0.0000	0.0000
	ite.	3.0	3.0	3.0	3.0	3.0

$$\boldsymbol{\lambda}^* = (-1.292714668049, 0.754908489475, 1.294574985726, 2.361040489862, \\ 8.801548359777, 17.222889574448, 35.134256281335, 783.036252731297)^T.$$

In this example, the initial point $\mathbf{c}^0$ is chosen by

$$\mathbf{c}^0 = \frac{\mathrm{floor}(\phi \times 10^{\psi} \cdot \mathbf{c}^*)}{\phi \times 10^{\psi}}.$$

We consider the following four cases: (a)$\phi = 5$ and $\psi = 1$; (b)$\phi = 3$ and $\psi = 2$; (c)$\phi = 1$ and $\psi = 2$; (d)$\phi = 1$ and $\psi = 3$. Here we take $B_0 = J(\mathbf{c}^0)^{-1}$ for the Ulm-like Cayley transform method. For both algorithms, the stopping tolerance for the outer (Newton) iterations is 10^{-10}. Table 3 displays the error of $\|\mathbf{c}^k - \mathbf{c}^*\|$ and the condition numbers $\kappa_2(J_k)$ of J_k for the above four initial points $\mathbf{c}^0$, where "ite." represents the number of outer iterations. We see from Table 3 that the outer iteration numbers of the Ulm-like Cayley transform method are comparable to that of the inexact Cayley transform method.

Table 3: $\|\mathbf{c}^k - \mathbf{c}^*\|$, ite. and $\kappa_2(J_k)$ of **Algorithms 2** and **3**

ini.	k	**Algorithm 2**					**Algorithm 3**	
		$\|\mathbf{c}^k - \mathbf{c}^*\|$				$\kappa_2(J_k)$	$\|\mathbf{c}^k - \mathbf{c}^*\|$	$\kappa_2(J_k)$
		$\beta = 1.5$	$\beta = 1.6$	$\beta = 1.8$	$\beta = 2.0$			
(a)	0	$3.3050e-2$	$3.3050e-2$	$3.3050e-2$	$3.3050e-2$	$1.4249e+3$	$3.3050e-2$	$1.4249e+3$
	1	$2.7831e-3$	$2.7831e-3$	$2.7831e-3$	$2.7831e-3$	$1.5447e+3$	$2.7831e-3$	$1.5447e+3$
	2	$7.0600e-5$	$7.0600e-5$	$7.0600e-5$	$7.0600e-5$	$1.5092e+3$	$4.0232e-5$	$1.5095e+3$
	3	$1.8498e-8$	$1.8497e-8$	$1.8497e-8$	$1.8497e-8$	$1.5098e+3$	$1.5346e-8$	$1.5098e+3$
	4	$3.0000e-14$	$3.0000e-14$	$3.0000e-14$	$3.0000e-14$		$4.0000e-14$	
	ite.	4	4	4	4		4	
(b)	0	$5.5304e-3$	$5.5304e-3$	$5.5304e-3$	$5.5304e-3$	$1.6134e+3$	$5.5304e-3$	$1.63e+1$
	1	$4.6484e-4$	$4.6484e-4$	$4.6484e-4$	$4.6485e-4$	$1.5064e+3$	$4.6485e-4$	$1.5064e+3$
	2	$4.8975e-7$	$4.8975e-7$	$4.8976e-7$	$4.8976e-7$	$1.5098e+3$	$2.7488e-6$	$1.5098e+3$
	3	$1.3300e-12$	$1.3300e-12$	$1.3000e-12$	$1.3200e-12$		$9.5070e-11$	
	ite.	3	3	3	3		3	
(c)	0	$1.3298e-2$	$1.3298e-2$	$1.3298e-2$	$1.3298e-2$	$1.7820e+3$	$1.3298e-2$	$1.7820e+3$
	1	$8.8146e-4$	$8.8146e-4$	$8.8146e-4$	$8.8146e-4$	$1.5214e+3$	$8.97e-1$	$1.5214e+3$
	2	$9.0149e-6$	$9.0149e-6$	$9.0149e-6$	$9.0149e-6$	$1.5098e+3$	$1.11e-1$	$1.5099e+3$
	3	$2.5774e-10$	$2.5765e-10$	$2.5765e-10$	$2.5766e-10$	$1.5098e+3$	$3.98e-3$	$1.5098e+3$
	4	$3.0000e-14$	$3.0000e-14$	$3.0000e-14$	$6.0000e-14$		$3.0000e-14$	
	ite.	4	4	4	4		4	
(d)	0	$1.3993e-3$	$1.3993e-3$	$1.3993e-3$	$1.3993e-3$	$1.5123e+3$	$1.3993e-3$	$1.5123e+3$
	1	$4.9801e-6$	$4.9801e-6$	$4.9817e-6$	$4.9817e-6$	$1.5099e+3$	$4.9817e-6$	$1.5099e+3$
	2	$1.7109e-10$	$1.7109e-10$	$1.7154e-10$	$1.7154e-10$	$1.5098e+3$	$3.5644e-10$	$1.5098e+3$
	3	$5.0000e-14$	$5.0000e-14$	$4.0000e-14$	$4.0000e-14$		$4.0000e-14$	
	ite.	3	3	3	3		3	

REFERENCES

1. Z. J. Bai, R. H. Chan and B. Morini, An inexact Cayley transform for inverse eigenvalue problem, *Inver. Problems*, **20** (2004), 1675-1689.

2. P. J. Brussard and P. W. Glaudemans, *Shell Model Applications in Nuclear Spectroscopy*, Elsevier, New York, 1977.

3. C. I. Byrnes, Pole placement by output feedback, Three Decades of Mathematics Systems Theory, in: *Lecture Notes in Control and Inform. Sci.*, Vol. 135, New York, Springer, 1989, 31-78.

4. R. H. Chan, H. L. Chung and S. F. Xu, The inexact Newton-like method for inverse eigenvalue problem, *BIT Numer. Math.*, **43** (2003), 7-20.

5. R. H. Chan, S. F. Xu and H. M. Zhou, On the convergence of a quasi-Newton method for inverse eigenvalue problem, *SIAM J. Numer. Anal.*, **36** (1999), 436-441.

6. M. T. Chu, Inverse eigenvalue problems, *SIAM Rev.*, **40** (1998), 1-39.

7. M. T. Chu and G. H. Golub, Structured inverse eigenvalue problems, *Acta Numer.*, **11** (2002), 1-71.

8. 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), 559-573.

9. S. Elhay and Y. M. Ram, An affine inverse eigenvalue problem, *Inverse Problems*, **18** (2002), 455-466.

10. J. A. Ezquerro and M. A. Hernández, The Ulm method under mild differentiability conditions, *Numer. Math.*, **109** (2008), 193-207.

11. R. W. Freund and N. M. Nachtigal, QMR: a quasi-minimal residual method for non-Hermitian linear systems, *Numer. Math.*, **60** (1991), 315-339.

12. 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), 634-667.

13. A. Galperin and Z. Waksman, Ulm's method under regular smoothness, *Numer. Funct. Anal. Optim.*, **19** (1998), 285-307.

14. G. M. L. Gladwell, Inverse problems in vibration, *Appl. Mech. Rev.*, **39** (1986), 1013-1018.

15. G. M. L. Gladwell, Inverse problems in vibration II, *Appl. Mech. Rev.*, **49** (1996), 25-34.

16. I. Gustafsson, A class of first order factorizations, *BIT*, **18** (1978), 142-156.

17. J. M. Gutiérrez, M. A. Hernández and N. Romero, A note on a modification of Moser's method, *Journal of Complexity*, **24** (2008), 185-197.

18. O. Hald, *On discrete and numerical Sturm-Liouville problems*, Ph.D. dissertation, Dept. Mathematics, New York University, New York, 1970.

19. O. Hald, On a Newton-Moser type method, *Numer. Math.*, **23** (1975), 411-426.

20. K. T. Joseph, Inverse eigenvalue problem in structural design, *AIAA J.*, **30** (1992), 2890-2896.

21. W. N. Kublanovskaja, *On an approach to the solution of the inverse eigenvalue problem*, Zap. Naucn. Sem. Leningrad. Otdel. Mat. Inst., in V. A. Steklova Akad. Nauk SSSR, 1970, 138-149.

22. C. Li and W. P. Shen, Local convergence of inexact methods under the Hölder condition, *J. Comput. Appl. Math.*, **222** (2008), 544-560.

23. N. Li, A matrix inverse eigenvalue problem and its application, *Linear Algebr. Appl.*, **266** (1997), 143-52.

24. J. Moser, *Stable and random motions in dynamical systems with special emphasis on celestian mechancis*, Herman Weil Lectures, Annals of Mathematics Studies, Vol. 77, Prinston Univ. Press, Prinston, New Jersey, 1973.

25. M. Müller, An inverse eigenvalue problem: computing B-stable Runge-Kutta methods having real poles, *BIT*, **32** (1992), 676-688.

26. R. L. Parker and K. A. Whaler, Numerical methods for establishing solutions to the inverse problem of electromagnetic induction, *J. Geophys. Res.*, **86** (1981), 9574-9584.

27. J. Peinado and A. M. Vidal, *A New Parallel Approach to the Toeplitz Inverse Eigenproblem Using Newton-Like Methods*, Lecture Notes in Computer Science (Berlin: Springer), 1981/2001, 355-368.

28. H. Petzeltova, Remark on Newton-Moser type method, *Commentat. Math. Univ. Carol.*, **21** (1980), 719-725.

29. F. A. Potra and V. Pták, Nondiscrete induction and an inversion-free modification of Newton's method, *Čas. Pro pěstováni Mat.*, **108** (1983), 333-341.

30. M. S. Ravi, J. Rosenthal and X. A. Wang, On decentralized dynamic pole placement and feedback stabilization IEEE Trans, *Automat. Control*, **40** (1995), 1603-1614.

31. W. P. Shen, C. Li and X. Q. Jin, An Ulm-like method for inverse eigenvalue problems, *Appl. Numer. Math.*, **61** (2011), 356-367.

32. W. F. Trench, Numerical solution of the inverse eigenvalue problem for real symmetric Toeplitz matrices, *SIAM J. Sci. Comput.*, **18** (1997), 1722-1736.

33. S. Ulm, On iterative methods with successive approximation of the inverse operator, *Izv. Akad. Nauk. Est. SSR*, **16** (1967), 403-411.

34. S. F. Xu, *An introduction to inverse algebric eigenvalue problems*, Peking University Press, 1998.

35. E. J. Zehnder, A remark about Newton's method, *Commun. Pure Appl. Math.*, **27** (1974), 361-366.

Weiping Shen
Department of Mathematics
Zhejiang Normal University
Jinhua 321004
P. R. China
E-mail: shenweiping@zjnu.cn

Chong Li
Department of Mathematics
Zhejiang University
Hangzhou, 310027
P. R. China
and
Department of Mathematics
College of Sciences
King Saud University
P. O. Box 2455, Riyadh 11451
Saudi Arabia
E-mail: cli@zju.edu.cn