

Document downloaded from:

<http://hdl.handle.net/10251/162860>

This paper must be cited as:

Dalvand, Z.; Hajarian, M.; Román Moltó, JE. (2020). An extension of the Cayley transform method for a parameterized generalized inverse eigenvalue problem. *Numerical Linear Algebra with Applications*. 27(6):1-24. <https://doi.org/10.1002/nla.2327>



The final publication is available at

<https://doi.org/10.1002/nla.2327>

Copyright John Wiley & Sons

Additional Information

An Extension of the Cayley Transform Method for a Parameterized Generalized Inverse Eigenvalue Problem

Zeynab Dalvand^a, Jose E. Roman^{b,*}

^a*Department of Mathematics, Faculty of Mathematical Sciences, Shahid Beheshti University, General Campus, Evin, Tehran 19839, Iran*

^b*D. Sistemes Informàtics i Computació, Universitat Politècnica de València, Camí de Vera, s/n, 46022 València, Spain*

Abstract

Since recent studies have shown that the Cayley transform method can be an effective iterative method for solving the inverse eigenvalue problem, in this work, we consider using an extension of it for solving a type of parameterized generalized inverse eigenvalue problem and prove its locally quadratic convergence. This type of inverse eigenvalue problem, which includes multiplicative and additive inverse eigenvalue problems, appears in many applications. Also, we consider the case where the given eigenvalues are multiple. In this case, we describe a modified problem that is not overdetermined, and discuss the extension of the Cayley transform method for this modified problem. Finally, to demonstrate the effectiveness of these algorithms, we present some numerical examples to show that the proposed methods are practical and efficient.

Keywords: parameterized generalized inverse eigenvalue problem, Cayley transform
2000 MSC: 65F18

1. Introduction

The inverse eigenvalue problem can appear in various forms depending on the applications [1, 2, 3, 4, 5]. The most common type is the parameterized inverse eigenvalue problem [6, 1]. In this paper, we study a kind of inverse eigenvalue problem, called parameterized generalized inverse eigenvalue problem.

We will use the following notations. The transpose and the determinant of a matrix A are represented, respectively, by the symbols A^T and $\det(A)$. I_n stands for the $n \times n$ identity matrix and A_{ij} denotes the elements of the matrix A . We will use the symbols $A > 0$ and $A \geq 0$, when the matrix A is positive definite and positive semi-definite, respectively. The symbol $\|\cdot\|$ represents the Euclidean norm for vectors or its corresponding induced matrix norm.

Generally, in the inverse eigenvalue problems, the given information contains either all or part of the eigenvalues or eigenvectors and the unknown parameters are the elements of a matrix or a matrix pencil [1]. The classes of inverse eigenvalue problems were recognized and categorized according to their specifications by Chu and Golub [1].

One of the categories of inverse eigenvalue problems, that due to numerous applications has always been a concern for researchers, is the parameterized case [7, 8]. A parameterized inverse eigenvalue problem can be defined as follows:

Definition 1. Let $A(c) = A(c_1, c_2, \dots, c_n)$ be a given $n \times n$ matrix whose entries are analytic functions of parameters (c_1, c_2, \dots, c_n) . Given n real numbers $\lambda_1, \lambda_2, \dots, \lambda_n$, the problem of finding $c \in \mathbb{R}^n$ such that the eigenvalue problem $A(c)x = \lambda x$ has the prescribed eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_n$, is called the parameterized inverse eigenvalue problem (PIEP).

*Corresponding author.

Email addresses: z_dalvand@sbu.ac.ir, zeynab.dalvand137@gmail.com (Zeynab Dalvand), jroman@dsic.upv.es (Jose E. Roman)

In many applications the matrix-valued function $A(c)$ belongs to an affine family, resulting in the special case of the PIEP in the form below [9]:

Problem 1. *Given $n + 1$ real symmetric $n \times n$ matrices A_k ($k = 0, 1, \dots, n$) and n real numbers $\lambda_1^* \leq \lambda_2^* \leq \dots \leq \lambda_n^*$, find parameters $c \in \mathbb{R}^n$ such that the matrix*

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

has the given eigenvalues $\{\lambda_i^\}_{i=1}^n$.*

If the eigenvalues of $A(c)$ are denoted by $\{\lambda_i(c)\}_{i=1}^n$ with $\lambda_1(c) \leq \dots \leq \lambda_n(c)$ and the given eigenvalues $\{\lambda_i^*\}_{i=1}^n$ are distinct, then it is possible to solve the problem by means of quadratically convergent Newton-like methods operating on the system of nonlinear equations

$$f(c) := [\lambda_1(c) - \lambda_1^*, \dots, \lambda_n(c) - \lambda_n^*]^T = 0. \quad (1.2)$$

The seminal paper by Friedland, Nocedal and Overton [9] describes three methods, that need to solve a linear system of equations at each Newton step, whose coefficient matrix is the Jacobian. The three methods differ in how the (approximate) Jacobian matrix is computed. The Jacobian matrix depends on the eigenvectors $\{q_i(c)\}_{i=1}^n$. Method I explicitly computes eigenvectors at each Newton step, Method II updates them with one step of inverse iteration, and Method III updates the eigenvector matrix $Q(c)$ by means of the matrix exponential and Cayley transforms. The main computational cost of the aforementioned methods is the solution of the linear systems of equations (with the Jacobian matrix and also associated with inverse iteration or the Cayley transform). Some authors have proposed to relax the accuracy with which these systems are solved, via iterative solvers, thus obtaining an inexact method. In this case, it is important to avoid *oversolving* by adjusting the tolerance of the inner loop so that the requested accuracy increases as the Newton process converges. In particular, Chan *et al.* [10] propose an inexact Newton-like method that solves inexactly both the Jacobian system and the inverse iteration update, while Bai *et al.* [11] advocate for the inexact Cayley method where only the Jacobian system is solved inexactly. An alternative to the inexact Newton-like method (with inverse iteration update) was proposed by Shen *et al.* [12], where the correction is not computed by solving a linear system but by Ulm's method, which involves a matrix recurrence using the Jacobian matrix. This idea was also applied to the case of the Cayley transform method [13].

More recently, Aishima [14] has proposed a quadratically convergent method based on matrix equations, which can be seen as an improved version of the Cayley transform method. The main advantage is that orthogonality of the approximate eigenvector matrix can be refined without the Cayley transform, resulting in a significantly cheaper computation.

The case of multiple eigenvalues is more subtle, but was already addressed in [9] with appropriate modifications of the three methods. Some of the above methods have also been adapted for the case of multiple eigenvalues: the inexact Cayley transform method in [15], the Ulm-like Cayley transform method in [16], and Aishima's method in [17].

On the more theoretical side, many studies have focused on existence theory for different categories of these problems [18, 19]. Sufficient conditions for the solvability of algebraic inverse eigenvalue problems by using Brouwer's fixed-point theorem and the topological degree have been presented in [20] and [21], respectively. In [22], sufficient conditions for a positive solution of the algebraic inverse eigenvalue problem have been provided. Also, Xu [19] investigated the sufficient conditions for the solvability of algebraic inverse eigenvalue problems.

A modification of Problem 1 is the case where the affine family (1.1) is defined with $m < n$ matrices only, in which case the problem may not have an exact solution but it is possible to solve the problem in the least-squares sense [23]. Another extension of PIEP is the parameterized generalized inverse eigenvalue problem, that by using another matrix-valued function $B(c) = B(c_1, c_2, \dots, c_n)$, can be defined as follows:

Definition 2. Let $(A(c), B(c))$ be a given matrix pencil whose entries are analytic functions of parameters (c_1, c_2, \dots, c_n) . Finding $c \in \mathbb{R}^n$ such that the generalized eigenvalue problem $A(c)x = \lambda B(c)x$ has the given eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_n$, is called the parameterized generalized inverse eigenvalue problem (PGIEP).

In this paper, we focus on a special case of PGIEP, where we need another set of $n + 1$ real symmetric $n \times n$ matrices $\{B_k\}_{k=0}^n$ so that

$$B(c) = B_0 + \sum_{k=1}^n c_k B_k. \quad (1.3)$$

We further assume that $B(c) > 0$ whenever $c \in \Omega$, an open subset of \mathbb{R}^n . In this case $\{\lambda_i(c)\}_{i=1}^n$ with $\lambda_1(c) \leq \dots \leq \lambda_n(c)$ denote the eigenvalues of the pencil $(A(c), B(c))$.

Problem 2. Given $A(c)$ (1.1) and $B(c)$ (1.3) with $B(c) > 0$ for $c \in \Omega$, and given n real numbers $\lambda_1^* \leq \dots \leq \lambda_n^*$, find $c \in \mathbb{R}^n$ such that $\lambda_i(c) = \lambda_i^*$ for $i = 1, \dots, n$.

Problem 2 appears in many practical applications in areas such as structural engineering, mechanics, and physics [3]. Examples of these applications are the structural design [24, 25], studying a vibrating string [26], nuclear spectroscopy, the educational testing problem [9], the graph partitioning problem, the design of control systems [22] and factor analysis [27]. These problems have also many applications in mathematical and numerical analysis, such as Sturm-Liouville problems and preconditioning.

For Problem 2, like in Problem 1, there is a lot of literature on the theory of solvability. In [7], Dai *et al.* stated the sufficient conditions for guaranteeing the existence of a solution for the parameterized generalized inverse eigenvalue problem. These conditions have been presented earlier for the parameterized inverse eigenvalue problem by Ji [28].

Problem 2 was approached by many authors. We next review the most important methods. Dai and Lancaster [29] extend Method I of Friedland *et al.* [9], that is, via the explicit computation of eigenvectors of the pencil $(A(c), B(c))$. Practically, this method requires computing the complete solution of the generalized eigenvalue problem $A(c)x = \lambda B(c)x$ in each iteration of Newton's method. On the other hand Shu *et al.* [30] introduced the Homotopy solution for the system of nonlinear equations (1.2).

The method of Aishima [14] mentioned above extends naturally to Problem 2. Dai *et al.* [7] also propose a method for Problem 2, but based on a different principle: smooth LU decomposition, which reformulates the nonlinear function (1.2). A variation of this latter method is presented in [31], by using a QR-like decomposition. In this method, the nonlinear function is written as

$$f(c) = \begin{bmatrix} r_{nn}^{(1)}(c) \\ r_{nn}^{(2)}(c) \\ \vdots \\ r_{nn}^{(n)}(c) \end{bmatrix}, \quad (1.4)$$

in which $r_{nn}^{(i)}(c)$ is obtained by computing a QR-like decomposition of $(A(c) - \lambda_i B(c))$ for $i = 1, 2, \dots, n$,

$$A(c) - \lambda_i B(c) = Q_i(c) R_i(c), \quad R_i(c) = \begin{bmatrix} R_{11}^{(i)}(c) & R_{12}^{(i)}(c) \\ 0 & r_{nn}^{(i)}(c) \end{bmatrix},$$

where $R_{11}^{(i)}(c)$ is upper triangular and $Q_i(c)$ is orthogonal.

In addition, Lancaster [32] and Biegler-Konig [33] presented a formulation based on determinant evaluation for the parameterized inverse eigenvalue problems. This formulation was used for PGIEP in [34]. In this method, the system of nonlinear equations is constructed as

$$f(c) = \begin{bmatrix} \det(A(c) - \lambda_1 B(c)) \\ \det(A(c) - \lambda_2 B(c)) \\ \vdots \\ \det(A(c) - \lambda_n B(c)) \end{bmatrix} = 0,$$

but it should be noted that this formulation is not computationally attractive.

Most of the methods to solve inverse eigenvalue problems have high computational complexity, because they need to solve an eigenvalue problem in each iteration. To avoid this and to reduce the computational cost, we propose a numerical algorithm based on the Cayley transform method.

In this paper, two extensions of the Cayley transform method are used to find an answer for a parameterized generalized inverse eigenvalue problem. The first one is for Problem 2 with distinct specified eigenvalues and the second one for the case where multiple eigenvalues are given. We also provide some convergence analysis that consists of extensions of ideas developed in [9, 11]. The paper concludes with some results of numerical experiments of both algorithms, in section 4.

2. Formulation

First, we review the approach proposed by Friedland *et al.* [9] based on the use of matrix exponential and Cayley transforms. Let Q be an orthogonal matrix, then a solution to Problem 1 can be described by c^* and Q as

$$Q^T A(c^*) Q = \Lambda^*, \quad (2.1)$$

$$Q^T Q = I, \quad (2.2)$$

where $\Lambda^* = \text{diag}(\lambda_1^*, \lambda_2^*, \dots, \lambda_n^*)$. Suppose that $Q^{(k)}$ is the current estimate of Q and $Z^{(k)}$ is a skew-symmetric matrix. Let us write $Q = Q^{(k)} e^{Z^{(k)}}$. Then, using the Taylor series of the exponential function and the definition of the skew-symmetric matrix, i.e., $Z^T = -Z$, we can express (2.1) as

$$\begin{aligned} Q^{(k)T} A(c^*) Q^{(k)} &= e^{Z^{(k)}} \Lambda^* e^{-Z^{(k)}} \\ &= (I + Z^{(k)} + \frac{1}{2} Z^{(k)2} + \dots) \Lambda^* (I - Z^{(k)} + \frac{1}{2} Z^{(k)2} + \dots) \\ &= \Lambda^* + Z^{(k)} \Lambda^* - \Lambda^* Z^{(k)} + \mathcal{O}(\|Z^{(k)}\|^2). \end{aligned}$$

Suppose that $c^{(k)}$ denotes the current approximation of c^* , it can be updated as $c^{(k+1)}$ by neglecting second-order terms in $Z^{(k)}$ as

$$Q^{(k)T} A(c^{(k+1)}) Q^{(k)} = \Lambda^* + Z^{(k)} \Lambda^* - \Lambda^* Z^{(k)}. \quad (2.3)$$

Assuming that $A(c)$ is defined by (1.1), i.e.,

$$A(c) = A_0 + \sum_{i=1}^n c_i A_i, \quad (2.4)$$

the parameters vector $c^{(k+1)}$ is obtained by equating the diagonal elements in (2.3) as

$$q_i^{(k)T} (A_0 + \sum_{j=1}^n c_j^{(k+1)} A_j) q_i^{(k)} = \lambda_i^*, \quad i = 1, 2, \dots, n. \quad (2.5)$$

By defining $d_i^{(k)} = q_i^{(k)T} A_0 q_i^{(k)}$ and $J_{ij}^{(k)} = q_i^{(k)T} A_j q_i^{(k)}$ for $i, j = 1, \dots, n$, we find $c^{(k+1)}$ by solving the linear system

$$J^{(k)} c^{(k+1)} = \lambda^* - d^{(k)}, \quad (2.6)$$

where

$$\lambda^* = [\lambda_1^*, \lambda_2^*, \dots, \lambda_n^*]^T.$$

Then, equating the off-diagonal elements in (2.3),

$$q_i^{(k)T} A(c^{(k+1)}) q_j^{(k)} = Z_{ij}^{(k)} (\lambda_j^* - \lambda_i^*), \quad i, j = 1, 2, \dots, n, \quad i \neq j, \quad (2.7)$$

ALGORITHM 1: Cayley transform method for Problem 1

Input: Problem matrices $\{A_k\}_{k=0}^n$, given eigenvalues $\Lambda^* = \text{diag}(\lambda_1^*, \lambda_2^*, \dots, \lambda_n^*)$ and initial guess $c^{(0)} \in \mathbb{R}^n$
Output: Computed solution $c^{(k+1)}$

- 1 Compute matrix $A(c^{(0)})$ and its orthonormal eigenvectors $[q_1(c^{(0)}), q_2(c^{(0)}), \dots, q_n(c^{(0)})]$
- 2 Let $Q^{(0)} = [q_1^{(0)}, q_2^{(0)}, \dots, q_n^{(0)}] = [q_1(c^{(0)}), q_2(c^{(0)}), \dots, q_n(c^{(0)})]$
- 3 **for** $k = 0, 1, 2, \dots$ **do**
- 4 **if** $\|Q^{(k)T} A(c^{(k)}) Q^{(k)} - \Lambda^*\|$ is sufficiently small **then** exit
- 5 Form vector $d^{(k)}$ and the approximate Jacobian matrix $J^{(k)}$

$$J_{ij}^{(k)} = q_i^{(k)T} A_j q_i^{(k)}$$

$$d_i^{(k)} = q_i^{(k)T} A_0 q_i^{(k)}$$
- 6 Solve the Jacobian equation (2.6) to obtain $c^{(k+1)}$
- 7 Compute the skew-symmetric matrix $Z^{(k)}$ using (2.8)
- 8 Compute $H^{(k)} = (I - \frac{1}{2}Z^{(k)})Q^{(k)T}$ and then solve the linear systems
$$(I + \frac{1}{2}Z^{(k)})v_i = h_i^{(k)}, \quad i = 1, 2, \dots, n \tag{2.10}$$

and set

$$Q^{(k+1)T} = [v_1, v_2, \dots, v_n] \tag{2.11}$$
- 9 **end**

and, assuming that the given eigenvalues are distinct, we obtain the off-diagonal elements of $Z^{(k)}$ as

$$Z_{ij}^{(k)} = \frac{q_i^{(k)T} A(c^{(k+1)}) q_j^{(k)}}{\lambda_j^* - \lambda_i^*}, \quad i, j = 1, 2, \dots, n, \quad i \neq j. \tag{2.8}$$

Finally, we create an orthogonal approximation of e^Z , by using the Cayley transform $e^Z \approx (I + \frac{1}{2}Z)(I - \frac{1}{2}Z)^{-1}$, and we calculate the new estimation of the matrix Q as

$$Q^{(k+1)} = Q^{(k)}(I + \frac{1}{2}Z^{(k)})(I - \frac{1}{2}Z^{(k)})^{-1}. \tag{2.9}$$

Algorithm 1 describes this method.

2.1. Distinct eigenvalues

In this sub-section, we present a new algorithm for Problem 2, by adapting the Cayley transform method.

Let (A, B) be a matrix pencil with matrix A symmetric and B symmetric positive definite, then there exists a real nonsingular matrix X such that $X^T A X$ is a real diagonal matrix and

$$X^T B X = I,$$

and eigenvalues of the matrix pencil (A, B) are equal to the diagonal elements of $X^T A X$ [35]. We will use the following result for a pencil of matrix-valued functions $(A(c), B(c))$.

Theorem 1 ([29]). *Let $\{A_i\}_{i=0}^n$ and $\{B_i\}_{i=0}^n$ be sets of real symmetric $n \times n$ matrices and suppose that the pencil of matrix-valued functions $(A(c), B(c))$ at $c = c^*$ has prescribed eigenvalues $\lambda_1^*, \lambda_2^*, \dots, \lambda_n^*$ and that $q_1^*, q_2^*, \dots, q_n^*$ are the corresponding eigenvectors so that*

$$[q_1^* \ q_2^* \ \dots \ q_n^*]^T A(c^*) [q_1^* \ q_2^* \ \dots \ q_n^*] = \text{diag}(\lambda_1^*, \lambda_2^*, \dots, \lambda_n^*),$$

$$[q_1^* \ q_2^* \ \dots \ q_n^*]^T B(c^*) [q_1^* \ q_2^* \ \dots \ q_n^*] = I.$$

There exists a neighbourhood of c^* in which the generalized eigenvalue problem for the pencil $(A(c), B(c))$ has real analytic eigenvalues $\lambda_1(c), \lambda_2(c), \dots, \lambda_n(c)$ and corresponding real analytic eigenvectors $q_1(c), q_2(c), \dots, q_n(c)$ and

$$\begin{aligned} Q(c)^T A(c) Q(c) &= \Lambda(c), \\ Q(c)^T B(c) Q(c) &= I, \end{aligned}$$

where $Q(c) = [q_1(c) \ q_2(c) \ \dots \ q_n(c)]$ and $\Lambda(c) = \text{diag}(\lambda_1(c), \lambda_2(c), \dots, \lambda_n(c))$. In addition

$$\frac{\partial \lambda_i(c)}{\partial c_j} = q_i(c)^T (A_j - \lambda_i^* B_j) q_i(c).$$

In particular, we see that if Problem 2 has a solution c^* , then there is a neighbourhood of c^* in which the eigenvalues of the pencil of matrix-valued functions $(A(c), B(c))$ are distinct and are differentiable functions.

According to Theorem 1, c and Q are a solution to Problem 2, if the following equations are satisfied:

$$\begin{cases} Q^T A(c) Q = \Lambda^*, \\ Q^T B(c) Q = I. \end{cases} \quad (2.12)$$

Suppose that $c^{(k)}$ and $Q^{(k)}$ represent our current estimation of c and Q , respectively. Similarly to the Cayley transform procedure, let us write $Q = Q^{(k)} e^Z$ where matrix Z , as opposed to Algorithm 1, is not skew-symmetric, then using (2.12), we have

$$\begin{cases} Q^{(k)T} A(c) Q^{(k)} = e^{-Z^T} \Lambda^* e^{-Z}, \\ Q^{(k)T} B(c) Q^{(k)} = e^{-Z^T} e^{-Z}. \end{cases}$$

Then, by using the Taylor series of e^{-Z} , we have

$$Q^{(k)T} A(c) Q^{(k)} = (I - Z^T + \frac{1}{2}(Z^T)^2 + \dots) \Lambda^* (I - Z + \frac{1}{2}Z^2 + \dots) = \Lambda^* - Z^T \Lambda^* - \Lambda^* Z + \mathcal{O}(\|Z\|^2), \quad (2.13a)$$

$$Q^{(k)T} B(c) Q^{(k)} = (I - Z^T + \frac{1}{2}(Z^T)^2 + \dots) (I - Z + \frac{1}{2}Z^2 + \dots) = I - Z^T - Z + \mathcal{O}(\|Z\|^2). \quad (2.13b)$$

Neglecting second order terms and subtracting (2.13b) post-multiplied by Λ^* from (2.13a) results in

$$Q^{(k)T} A(c) Q^{(k)} - Q^{(k)T} B(c) Q^{(k)} \Lambda^* = Z \Lambda^* - \Lambda^* Z. \quad (2.14)$$

To improve our approximations to the parameters $c^{(k+1)}$, equating the diagonal elements of (2.14), we get

$$q_i^{(k)T} (A_0 + \sum_{j=1}^n c_j^{(k+1)} A_j) q_i^{(k)} - q_i^{(k)T} (\lambda_i^* B_0 + \lambda_i^* \sum_{j=1}^n c_j^{(k+1)} B_j) q_i^{(k)} = 0. \quad (2.15)$$

Let us write

$$J_{ij}^{(k)} = q_i^{(k)T} (A_j - \lambda_i^* B_j) q_i^{(k)}, \quad (2.16)$$

$$d_i^{(k)} = q_i^{(k)T} (A_0 - \lambda_i^* B_0) q_i^{(k)}, \quad (2.17)$$

therefore, from (2.15), $c^{(k+1)}$ can be found from

$$J^{(k)} c^{(k+1)} = -d^{(k)}.$$

By equating the diagonal elements of the sum of (2.13a) and (2.13b), and the off-diagonal elements of (2.14), we obtain the elements of matrix $Z^{(k)}$ as

$$Z_{ij}^{(k)} = \begin{cases} \frac{q_i^{(k)T} A(c^{(k+1)}) q_i^{(k)} + q_i^{(k)T} B(c^{(k+1)}) q_i^{(k)} - \lambda_i^* - 1}{-(2+2\lambda_i^*)}, & i = 1, 2, \dots, n, \quad i = j, \\ \frac{q_i^{(k)T} A(c^{(k+1)}) q_j^{(k)} - \lambda_j^* q_i^{(k)T} B(c^{(k+1)}) q_j^{(k)}}{\lambda_j^* - \lambda_i^*}, & i = 1, 2, \dots, n, \quad j = 1, 2, \dots, n, \quad j \neq i. \end{cases} \quad (2.18)$$

ALGORITHM 2: Cayley transform method for Problem 2

Input: Problem matrices $\{A_k\}_{k=0}^n$ and $\{B_k\}_{k=0}^n$, given eigenvalues $\Lambda^* = \text{diag}(\lambda_1^*, \lambda_2^*, \dots, \lambda_n^*)$ and initial guess $c^{(0)} \in \mathbb{R}^n$

Output: Computed solution $c^{(k+1)}$

- 1 Compute the matrix pencil $(A(c^{(0)}), B(c^{(0)}))$ and its eigenvectors $[q_1(c^{(0)}), q_2(c^{(0)}), \dots, q_n(c^{(0)})]$
- 2 Let $Q^{(0)} = [q_1^{(0)}, q_2^{(0)}, \dots, q_n^{(0)}] = [q_1(c^{(0)}), q_2(c^{(0)}), \dots, q_n(c^{(0)})]$
- 3 **for** $k = 0, 1, \dots$ **do**
- 4 **if** $\max\{\|Q^{(k)T} A(c^{(k)}) Q^{(k)} - \Lambda^*\|, \|Q^{(k)T} B(c^{(k)}) Q^{(k)} - I\|\}$ *is small enough* **then** exit
- 5 Compute the Jacobian matrix $J^{(k)}$ and $d^{(k)}$ using the equations (2.16) and (2.17), respectively
- 6 Solve $c^{(k+1)}$ from the Jacobian equation
$$J^{(k)} c^{(k+1)} = -d^{(k)}$$
- 7 Compute $A(c^{(k+1)})$ and $B(c^{(k+1)})$
- 8 Find $Z^{(k)}$ using (2.18)
- 9 Compute the subsequent approximation of the matrix of eigenvectors $Q^{(k+1)}$ using (2.19)
- 10 **end**

Using the Cayley transform, we compute the subsequent approximation of the matrix of eigenvectors $Q^{(k+1)}$, as

$$Q^{(k+1)} = Q^{(k)} e^{Z^{(k)}} \simeq Q^{(k)} \left(I + \frac{1}{2} Z^{(k)} \right) \left(I - \frac{1}{2} Z^{(k)} \right)^{-1}. \quad (2.19)$$

Thus the Cayley transform method for solving Problem 2 can be summarized in Algorithm 2.

2.2. Multiple eigenvalues

Since the inverse eigenvalue problem with a multiple eigenvalue is overdetermined, Friedland *et al.* [9] presented a modified problem to remove this feature, by using only $n - s$ smallest eigenvalues

$$\lambda_1^* = \dots = \lambda_t^* < \lambda_{t+1}^* < \dots < \lambda_{n-s}^*, \quad (2.20)$$

where $s = \frac{t(t-1)}{2}$. This modified problem has been extended to Problem 2 in [31]. The rationale for using $n - s$ smallest eigenvalues is as follows. The two relations of (2.12) can be seen as a system of $n(n+1)$ equations (each of the two parts has $n(n+1)/2$ due to symmetry), where the unknowns are the n^2 elements of Q and the n entries of c . However, if a multiple eigenvalue is present, s degrees of freedom are lost in the first part of (2.12) since they just describe coordinate transformations in the eigenspace of λ_1 . Hence, either s of the remaining eigenvalues are not specified, or s parameters must be added. Therefore, by accepting the above assumption, the modified problem is expressed as follows:

Problem 3. Find the parameters $c = (c_1, c_2, \dots, c_n)^T$ such that the matrix pencil $(A(c), B(c))$ has the given numbers $\lambda_1^* = \dots = \lambda_t^* < \lambda_{t+1}^* < \dots < \lambda_{n-s}^*$ for its $n - s$ smallest eigenvalues.

Since in Problem 3 s of the rows and columns of the Jacobian matrix (2.16) have effectively been removed, Algorithm 2 must now be adapted for solving this problem. To do this, we first focus on solving the following matrix equations (with second order terms neglected), as before

$$\begin{cases} Q^{(k)T} A(c) Q^{(k)} &= \Lambda^* - Z^T \Lambda^* - \Lambda^* Z + \mathcal{O}(\|Z\|^2), \\ Q^{(k)T} B(c) Q^{(k)} &= I - Z - Z^T + \mathcal{O}(\|Z\|^2). \end{cases} \quad (2.21)$$

As formerly, we divide the computations of c and Z , for solving this system of equations. But, it is clear that $s = t(t-1)/2$ of the Z_{ij} obtained by (2.18), namely those for which $1 \leq i \leq j \leq t$, are of no help in solving (2.21) and may be removed from the equation, since these eigenvalues are multiple. Also, we can see that

the system of equations (2.21) is appropriate to specify only $n - s$ eigenvalues, since only $n - s$ eigenvalues are given in Problem 3. Thus, we must replace Λ^* in (2.21) by $\bar{\Lambda}^* = \text{diag}(\bar{\lambda}_1^*, \bar{\lambda}_2^*, \dots, \bar{\lambda}_{n-s}^*, \bar{\lambda}_{n-s+1}^*, \dots, \bar{\lambda}_n^*)$, where $\bar{\lambda}_i^* = \lambda_i^*$ for $i = 1, 2, \dots, n - s$, and the last s entries are free parameters. For $i = 1, 2, \dots, n - s$, we can obtain $c^{(k+1)}$ from the diagonal elements of (2.21), omitting the quadratic terms,

$$\begin{cases} q_i^{(k)T} A(c^{(k+1)}) q_i^{(k)} &= \lambda_i^* - 2\lambda_i^* Z_{ii}, \\ q_i^{(k)T} B(c^{(k+1)}) q_i^{(k)} &= 1 - 2Z_{ii}, \end{cases} \quad (2.22)$$

and, subtracting the second part multiplied by λ_i^* from the first part and using the definition of $A(c)$ and $B(c)$, we have

$$q_i^{(k)T} (A_0 + \sum_{j=1}^n c_j^{(k+1)} A_j) q_i^{(k)} - q_i^{(k)T} (\lambda_i^* B_0 + \sum_{j=1}^n \lambda_i^* c_j^{(k+1)} B_j) q_i^{(k)} = 0.$$

Therefore, the new expression for computing $c^{(k+1)}$ has two parts. The first equations, for $i = 1, \dots, n - s$, take the form

$$q_i^{(k)T} \left(\sum_{\ell=1}^n (A_\ell - \lambda_i^* B_\ell) c_\ell^{(k+1)} \right) q_i^{(k)} = -q_i^{(k)T} (A_0 - \lambda_i^* B_0) q_i^{(k)}, \quad (2.23)$$

while for the remaining ones, with $1 \leq i \leq j \leq t$, by using the off-diagonal elements of (2.21), we have

$$q_i^{(k)T} \left(\sum_{\ell=1}^n (A_\ell - \lambda_i^* B_\ell) c_\ell^{(k+1)} \right) q_j^{(k)} = -q_i^{(k)T} (A_0 - \lambda_i^* B_0) q_j^{(k)}. \quad (2.24)$$

Previously, equations (2.24) were not enforced by Algorithm 1; they were not needed since the given eigenvalues were sufficient. We can write the combined system (2.23)–(2.24) as

$$K^{(k)} c^{(k+1)} = b^{(k)}. \quad (2.25)$$

Now, after obtaining $c^{(k+1)}$, we need to clarify how the unknowns Z_{ij} are to be calculated. According to the off-diagonal entries of (2.21) and using $\bar{\Lambda}^* = \text{diag}(\bar{\lambda}_1^*, \bar{\lambda}_2^*, \dots, \bar{\lambda}_n^*)$, for any $i \neq j$ we have

$$q_i^{(k)T} A(c^{(k+1)}) q_j^{(k)} = -\bar{\lambda}_j^* Z_{ji}^{(k)} - \bar{\lambda}_i^* Z_{ij}^{(k)}, \quad (2.26)$$

and

$$q_i^{(k)T} B(c^{(k+1)}) q_j^{(k)} = -Z_{ji}^{(k)} - Z_{ij}^{(k)}, \quad (2.27)$$

where

$$\bar{\lambda}_i^* = \begin{cases} \lambda_j^* & \text{if } i = 1, 2, \dots, n - s, \\ q_i^{(k)T} A(c^{(k+1)}) q_i^{(k)} & \text{otherwise.} \end{cases} \quad (2.28)$$

Therefore, for $i \neq j$ such that $\bar{\lambda}_i^*$ and $\bar{\lambda}_j^*$ satisfy

$$|\bar{\lambda}_i^* - \bar{\lambda}_j^*| > \epsilon,$$

where ϵ is a small multiple of the machine precision, Z_{ij} is produced uniquely as

$$Z_{ij}^{(k)} = \frac{q_i^{(k)T} A(c^{(k+1)}) q_j^{(k)} - \bar{\lambda}_j^* q_i^{(k)T} B(c^{(k+1)}) q_j^{(k)}}{\bar{\lambda}_j^* - \bar{\lambda}_i^*}. \quad (2.29)$$

Further, to handle (2.26) and (2.27) for multiple eigenvalues $\bar{\lambda}_i^* = \bar{\lambda}_j^*$, we set $Z_{ij}^{(k)} = 0$, when this choice substituted into the equations, satisfies both equations at the same time.

Finally, we calculate the diagonal elements of matrix $Z^{(k)}$ corresponding to the alternative eigenvalues $\bar{\lambda}_i^*$. Equating the off-diagonal elements of (2.22), we get

$$Z_{ii}^{(k)} = \frac{q_i^{(k)T} A(c^{(k+1)}) q_i^{(k)} + q_i^{(k)T} B(c^{(k+1)}) q_i^{(k)} - \bar{\lambda}_i^* - 1}{-(2 + 2\bar{\lambda}_i^*)}, \quad i > t. \quad (2.30)$$

The approximation of the matrix of eigenvectors is computed with the Cayley transform, as before. A final detail is required to complete the description of the method. Since matrix $Z^{(k)}$ is not assumed to be skew-symmetric, the next iterate $Q^{(k+1)}$ may depart from B -orthogonality. Hence, once $Q^{(k+1)}$ is computed, we *refine* it so that B -orthogonality is maintained. We use the Newton-Schulz iteration [36, section 8.3] derived from applying Newton's method to the equation $Q^{(k+1)T} B(c) Q^{(k+1)} = I$. Since this iteration is quadratically convergent and the initial guess $Q^{(k+1)}$ is already close to the solution, we apply just one iteration, resulting in

$$Q^{(k+1)} = \frac{1}{2} \hat{Q}^{(k+1)} \left(3I - \hat{Q}^{(k+1)T} B(c^{(k+1)}) \hat{Q}^{(k+1)} \right), \quad (2.31)$$

where

$$\hat{Q}^{(k+1)} = Q^{(k)} \left(I + \frac{1}{2} Z^{(k)} \right) \left(I - \frac{1}{2} Z^{(k)} \right)^{-1}.$$

Due to the analysis offered above, we can describe the Cayley transform method adapted for solving Problem 3 as shown in Algorithm 3.

3. Convergence analysis

The purpose of this section is to show the quadratic convergence of the Cayley transform method for solving Problems 2 and 3. We first show the convergence of Algorithm 2. To start, we set the following assumptions.

1. There exists c^* such that the given eigenvalues $\{\lambda_1^*, \lambda_2^*, \dots, \lambda_n^*\}$ are the eigenvalues for the matrix pencil $(A(c^*), B(c^*))$.
2. We consider that all the given eigenvalues are distinct.

We will now provide some introductory results that will be required for the convergence proofs.

Lemma 1. *For two matrix-valued functions (1.1) and (1.3), there exist two positive constants α_1 and α_2 such that for any c and \bar{c}*

$$\|A(c) - A(\bar{c})\| \leq \alpha_1 \|c - \bar{c}\|, \quad \|B(c) - B(\bar{c})\| \leq \alpha_2 \|c - \bar{c}\|.$$

Proof. By definition of $A(c)$ and $B(c)$ and let

$$\alpha_1 := \left(\sum_{l=1}^n \|A_l\|^2 \right)^{\frac{1}{2}}, \quad \alpha_2 := \left(\sum_{l=1}^n \|B_l\|^2 \right)^{\frac{1}{2}}.$$

□

Lemma 2. *If c^* is a solution for Problem 2 and Q is the matrix of the eigenvectors of the matrix pencil $(A(c^*), B(c^*))$ normalized so that $Q^T B(c^*) Q = I$, then we have*

$$\|q_i\| \leq \sqrt{\|B(c^*)^{-1}\|}, \quad i = 1, 2, \dots, n,$$

where q_i , $i = 1, 2, \dots, n$, are the columns of Q .

Proof. The result can be derived trivially from $1 = \|q_i^T B(c^*) q_i\| = \|B(c^*)^{-1} B(c^*)\|$. □

ALGORITHM 3: Modified Cayley transform method for Problem 3

Input: Problem matrices $\{A_k\}_{k=0}^n$ and $\{B_k\}_{k=0}^n$, given eigenvalues $\Lambda^* = \text{diag}(\lambda_1^*, \lambda_2^*, \dots, \lambda_{n-s}^*)$ and initial guess $c^{(0)} \in \mathbb{R}^n$

Output: Computed solution $c^{(k+1)}$

- 1 Compute the eigen-decomposition of the matrix pencil $(A(c^{(0)}), B(c^{(0)}))$ to obtain the $B(c^{(0)})$ -orthonormal eigenvectors $[q_1(c^{(0)}), q_2(c^{(0)}), \dots, q_n(c^{(0)})]$
- 2 Let $Q^{(0)} = [q_1^{(0)}, q_2^{(0)}, \dots, q_n^{(0)}] = [q_1(c^{(0)}), q_2(c^{(0)}), \dots, q_n(c^{(0)})]$
- 3 **for** $k = 0, 1, \dots$ **do**
- 4 **if for** $Q_{n-s}^{(k)} = \{q_1^{(k)}, q_2^{(k)}, \dots, q_{n-s}^{(k)}\}$ and $\Lambda_{n-s}^* = \text{diag}(\lambda_1^*, \lambda_2^*, \dots, \lambda_{n-s}^*)$,

$$\max \{ \|Q_{n-s}^{(k)T} A(c^{(k+1)}) Q_{n-s}^{(k)} - \Lambda_{n-s}^*\|, \|Q_{n-s}^{(k)T} B(c^{(k+1)}) Q_{n-s}^{(k)} - I_{n-s}\| \},$$
is small enough then exit
- 5 Compute the Jacobian matrix $K^{(k)}$ and $b^{(k)}$ using the equations (2.23) and (2.24)
- 6 Form $A(c^{(k+1)})$ and $B(c^{(k+1)})$
- 7 Compute the matrix $Z^{(k)}$ as: For each $1 \leq i, j \leq n$ such that $i \neq j$

$$Z_{ij}^{(k)} = \begin{cases} \frac{q_i^{(k)T} A(c^{(k+1)}) q_j^{(k)} - \bar{\lambda}_j^* q_i^{(k)T} B(c^{(k+1)}) q_j^{(k)}}{\lambda_i^* - \bar{\lambda}_j^*} & \text{if } |\bar{\lambda}_i^* - \bar{\lambda}_j^*| > \epsilon \\ 0 & \text{otherwise,} \end{cases}$$
 and for $i = j$,

$$Z_{ii}^{(k)} = \begin{cases} \frac{q_i^{(k)T} A(c^{(k+1)}) q_i^{(k)} + q_i^{(k)T} B(c^{(k+1)}) q_i^{(k)} - \bar{\lambda}_i^* - 1}{-(2+2\lambda_i^*)} & \text{if } i > t \\ 0 & \text{if } i \leq t \end{cases}$$
- 8 Compute the subsequent approximation of the matrix of eigenvectors $\hat{Q}^{(k+1)}$, using (2.19);

$$\hat{Q}^{(k+1)} = Q^{(k)} (I + \frac{1}{2} Z^{(k)}) (I - \frac{1}{2} Z^{(k)})^{-1}$$
 and replace it by the corresponding $B(c^{(k+1)})$ -orthogonal matrix, using one step of Newton-Schulz iteration

$$Q^{(k+1)} = \frac{1}{2} \hat{Q}^{(k+1)} (3I - \hat{Q}^{(k+1)T} B(c^{(k+1)}) \hat{Q}^{(k+1)})$$
- 9 **end**

Lemma 3. Suppose that the Jacobian $J^* = J(c^*)$ is nonsingular. Then there exist positive numbers δ_1 and ρ_1 such that, if $\|Z^{(k)}\| \leq \delta_1$ for some k , then $J^{(k)}$ is nonsingular and

$$\|(J^{(k)})^{-1}\| \leq \rho_1.$$

Proof. Using the definition of $J^{(k)}$ we have

$$|J_{ij}^{(k)} - J_{ij}^*| = |q_i^{(k)T} (A_j - \lambda_i^* B_j) q_i^{(k)} - q_i^T (A_j - \lambda_i^* B_j) q_i|, \quad (3.1)$$

and in view of the fact that $Q^{(k)} = Q e^{Z^{(k)}} = Q(I - Z^{(k)} + \dots)$, we can write

$$q_i^{(k)} \simeq q_i - \sum_{r=1}^n Z_{ri}^{(k)} q_r.$$

Therefore, we have

$$\begin{aligned}
|J_{ij}^{(k)} - J_{ij}^*| &= |(q_i - \sum_{r=1}^n Z_{ri}^{(k)} q_r)^T (A_j - \lambda_i^* B_j) (q_i - \sum_{r=1}^n Z_{ri}^{(k)} q_r) - q_i^T (A_j - \lambda_i^* B_j) q_i| \\
&= | -q_i^T (A_j - \lambda_i^* B_j) \sum_{r=1}^n Z_{ri}^{(k)} q_r - \sum_{r=1}^n Z_{ri}^{(k)} q_r^T (A_j - \lambda_i^* B_j) q_i + \sum_{r=1}^n Z_{ri}^{(k)} q_r^T (A_j - \lambda_i^* B_j) \sum_{r=1}^n Z_{ri}^{(k)} q_r | \\
&\leq |q_i^T (A_j - \lambda_i^* B_j) \sum_{r=1}^n Z_{ri}^{(k)} q_r| + | \sum_{r=1}^n Z_{ri}^{(k)} q_r^T (A_j - \lambda_i^* B_j) q_i | + | \sum_{r=1}^n Z_{ri}^{(k)} q_r^T (A_j - \lambda_i^* B_j) \sum_{r=1}^n Z_{ri}^{(k)} q_r |.
\end{aligned}$$

Therefore, by the Cauchy-Schwarz inequality, since $\|q_i\|^2 \leq \mu$ with $\mu = \|B(c^*)^{-1}\|$ and $\|\sum_{r=1}^n Z_{ri}^{(k)} q_r\| \leq \|Z^{(k)}\| \sqrt{\mu}$, we obtain

$$\begin{aligned}
|J_{ij}^{(k)} - J_{ij}^*| &\leq \sqrt{\mu} \| (A_j - \lambda_i^* B_j) \| \|Z^{(k)}\| \sqrt{\mu} \\
&\quad + \|Z^{(k)}\| \sqrt{\mu} \| (A_j - \lambda_i^* B_j) \| \sqrt{\mu} \\
&\quad + \|Z^{(k)}\| \sqrt{\mu} \| (A_j - \lambda_i^* B_j) \| \|Z^{(k)}\| \sqrt{\mu} \\
&= \mu \| (A_j - \lambda_i^* B_j) \| \|Z^{(k)}\| (2 + \|Z^{(k)}\|), \quad 1 \leq i, j \leq n.
\end{aligned}$$

Then, by using the Frobenius norm, we have

$$\|J^{(k)} - J^*\| \leq \|J^{(k)} - J^*\|_F \leq \sqrt{n} \mu \sqrt{\sum_{j=1}^n \| (A_j - \lambda_\omega^* B_j) \|^2 \|Z^{(k)}\| (2 + \|Z^{(k)}\|)}, \quad (3.2)$$

where $\sum_{j=1}^n \| (A_j - \lambda_\omega^* B_j) \| = \max_{1 \leq i \leq n} \sum_{j=1}^n \| (A_j - \lambda_i^* B_j) \|$.

Since J^* is nonsingular, using (3.2) and the continuity of the inverse matrix function, we conclude that $(J^{(k)})^{-1}$ exists and is uniformly bounded. This completes the proof. \square

Remark 1. It is possible to derive an expression for the bound of Lemma 3. Using Weyl's inequality for singular values we have

$$|\sigma_n(J^{(k)}) - \sigma_n(J^*)| \leq \|J^{(k)} - J^*\|, \quad (3.3)$$

where σ_n is the smallest singular value. Noting that $\|(J^{(k)})^{-1}\|^{-1} = \sigma_n(J^{(k)})$ and using (3.2) and (3.3) we can see

$$\| (J^{(k)})^{-1} \|^{-1} - \| (J^*)^{-1} \|^{-1} = |\sigma_n(J^{(k)}) - \sigma_n(J^*)| \leq \sqrt{n} \mu \sqrt{\sum_{j=1}^n \| (A_j - \lambda_\omega^* B_j) \|^2 \|Z^{(k)}\| (2 + \|Z^{(k)}\|)}.$$

It then follows that

$$\| (J^{(k)})^{-1} \|^{-1} \geq \| (J^*)^{-1} \|^{-1} - \sqrt{n} \mu \sqrt{\sum_{j=1}^n \| (A_j - \lambda_\omega^* B_j) \|^2 \|Z^{(k)}\| (2 + \|Z^{(k)}\|)},$$

and finally, we obtain

$$\| (J^{(k)})^{-1} \| \leq \frac{\| (J^*)^{-1} \|}{1 - \sqrt{n} \mu \| (J^*)^{-1} \| \|Z^{(k)}\| (2 + \|Z^{(k)}\|) \sqrt{\sum_{j=1}^n \| (A_j - \lambda_\omega^* B_j) \|}}, \quad (3.4)$$

where

$$\sqrt{n} \mu \| (J^*)^{-1} \| \sqrt{\sum_{j=1}^n \| (A_j - \lambda_\omega^* B_j) \|} \geq 1,$$

because $1 = \|I\| = \|J^* (J^*)^{-1}\| \leq \|J^*\| \| (J^*)^{-1} \|$.

Also, according to the previous lemmas and since all eigenvalues, and the eigenvectors corresponding to distinct eigenvalues, are Lipschitz continuous functions in a neighborhood of c^* , we present convergence results for Algorithms 2 and 3.

Theorem 2. *There exists θ such that, if $\|Z^{(k)}\| \leq \theta$ for some k , then the sequence of norms of the matrix $Z^{(k+1)}$ of Algorithm 2 converges to zero, and the iterates $\{c^{(k)}\}$ of Algorithm 2 quadratic converge to c^* .*

Proof. The matrix $\bar{Z}^{(k)}$ and vector $c^{(k+1)}$ of Algorithm 2 are defined by

$$\begin{cases} Q^{(k)T} A(c^{(k+1)}) Q^{(k)} &= \Lambda^* - \bar{Z}^{(k)T} \Lambda^* - \Lambda^* \bar{Z}^{(k)}, \\ Q^{(k)T} B(c^{(k+1)}) Q^{(k)} &= I - \bar{Z}^{(k)} - \bar{Z}^{(k)T}. \end{cases} \quad (3.5)$$

By subtracting the above equations from (2.13), we obtain

$$\begin{cases} Q^{(k)T} (A(c^*) - A(c^{(k+1)})) Q^{(k)} &= -Z^{(k)T} \Lambda^* - \Lambda^* Z^{(k)} + \bar{Z}^{(k)T} \Lambda^* + \Lambda^* \bar{Z}^{(k)} + \mathcal{O}(\|Z^{(k)}\|^2), \\ Q^{(k)T} (B(c^*) - B(c^{(k+1)})) Q^{(k)} &= -Z^{(k)} - Z^{(k)T} + \bar{Z}^{(k)} + \bar{Z}^{(k)T} + \mathcal{O}(\|Z^{(k)}\|^2). \end{cases} \quad (3.6)$$

Now, using Lemma 1, the diagonal equalities in (3.6) and the definition of the Jacobian (2.16), we have

$$\|c^* - c^{(k+1)}\| \leq 2\sqrt{n} \|Z^{(k)}\|^2 \|\Lambda^*\| \|J^{(k)}\|^{-1}, \quad (3.7)$$

and by using the off-diagonal elements in (3.6), we have, for $i \neq j$,

$$|\bar{Z}_{ij}^{(k)} - Z_{ij}^{(k)}| \leq \frac{|q_j^{(k)T} \sum_{l=1}^n (c_l^* - c_l^{(k+1)}) (A_l - \lambda_j^* B_l) q_i^{(k)}| + 2|\lambda_j^*| \|Z^{(k)}\|^2}{|\lambda_i^* - \lambda_j^*|}. \quad (3.8)$$

Also, noting that $\|q_i^{(k)}\|^2 \leq \mu$ with $\mu = \|B(c^*)^{-1}\|$, $Q^{(k)} = Qe^{-Z^{(k)}} = Q(I - Z^{(k)} + \dots)$, and using Lemma 1 we have

$$|q_j^{(k)T} \sum_{l=1}^n (c_l^* - c_l^{(k+1)}) (A_l - \lambda_j^* B_l) q_i^{(k)}| \leq \|c^* - c^{(k+1)}\| \left\| \sum_{l=1}^n (A_l - \lambda_j^* B_l) \right\| (1 + \|Z^{(k)}\|) \mu (1 + \|Z^{(k)}\|). \quad (3.9)$$

Thus, by applying (3.9) and (3.8), we obtain

$$|\bar{Z}_{ij}^{(k)} - Z_{ij}^{(k)}| \leq \frac{\|c^* - c^{(k+1)}\| \sqrt{\left\| \sum_{l=1}^n (A_l - \lambda_j^* B_l) \right\|} (1 + \|Z^{(k)}\|) \mu (1 + \|Z^{(k)}\|) + 2|\lambda_j^*| \|Z^{(k)}\|^2}{|\lambda_i^* - \lambda_j^*|}, \quad (3.10)$$

Similarly, the diagonal elements $|\bar{Z}_{ii}^{(k)} - Z_{ii}^{(k)}|$ are obtained. Using the inequality (3.7) and Lemma 3, and defining $\rho = \sqrt{n} \mu \|(J^*)^{-1}\| \sqrt{\left\| \sum_{l=1}^n (A_l - \lambda_\omega^* B_l) \right\|}$, we can see

$$\begin{aligned} |\bar{Z}_{ij}^{(k)} - Z_{ij}^{(k)}| &\leq \frac{2\|\Lambda^*\| \|Z^{(k)}\|^2 (1 - \rho(2 + \|Z^{(k)}\|) \|Z^{(k)}\| + \rho(1 + \|Z^{(k)}\|)^2)}{\delta(\lambda^*) (1 - \rho(2 + \|Z^{(k)}\|) \|Z^{(k)}\|)} \\ &\leq \frac{2\|\Lambda^*\| \|Z^{(k)}\|^2 (1 + \rho)}{\delta(\lambda^*) (1 - \rho(2 + \|Z^{(k)}\|) \|Z^{(k)}\|)}, \end{aligned}$$

where $\delta(\lambda^*) = \min_{i \neq j} |\lambda_i^* - \lambda_j^*|$. Also we know that $\|\bar{Z}^{(k)} - Z^{(k)}\| \leq \sqrt{\sum_{i,j} |\bar{Z}_{ij}^{(k)} - Z_{ij}^{(k)}|^2}$. Therefore, we obtain

$$\|\bar{Z}^{(k)} - Z^{(k)}\| \leq \frac{2n \|\Lambda^*\| \|Z^{(k)}\|^2 (1 + \rho)}{\delta(\lambda^*) (1 - \rho(2 + \|Z^{(k)}\|) \|Z^{(k)}\|)}. \quad (3.11)$$

Now suppose that $\theta = \frac{\delta(\lambda^*)}{8n\|\Lambda^*\|(1+\rho)}$, and due to this assumption, for $\rho \geq 1$ and $n \geq 2$, we see $\|Z^{(k)}\| \leq \theta \leq \frac{1}{16}$, and so

$$\|\bar{Z}^{(k)}\| \leq \|Z^{(k)}\| + \frac{2n\|\Lambda^*\|\|Z^{(k)}\|^2(1+\rho)}{\delta(\lambda^*)(1-\rho(2+\|Z^{(k)}\|)\|Z^{(k)}\|)}, \quad (3.12)$$

and $\|\bar{Z}^{(k)}\| \leq 1$ and therefore

$$\|(I - \frac{1}{2}\bar{Z}^{(k)})^{-1}\| \leq \frac{1}{1 - \frac{1}{2}\|\bar{Z}^{(k)}\|} \leq 2. \quad (3.13)$$

Let us now look at the updated matrix

$$Q^{(k+1)} = Qe^{-Z^{(k+1)}} \simeq Q(I - Z^{(k+1)}) = Q^{(k)}(I + \frac{1}{2}\bar{Z}^{(k)})(I - \frac{1}{2}\bar{Z}^{(k)})^{-1} = Q(I - Z^{(k)})(I + \frac{1}{2}\bar{Z}^{(k)})(I - \frac{1}{2}\bar{Z}^{(k)})^{-1},$$

we can see that

$$(I - Z^{(k+1)}) = (I - Z^{(k)})(I + \frac{1}{2}\bar{Z}^{(k)})(I - \frac{1}{2}\bar{Z}^{(k)})^{-1}.$$

From here we have

$$\begin{aligned} Z^{(k+1)} &= I - (I - Z^{(k)})(I + \frac{1}{2}\bar{Z}^{(k)})(I - \frac{1}{2}\bar{Z}^{(k)})^{-1} \\ &= (I - \frac{1}{2}\bar{Z}^{(k)})^{-1}[I - \frac{1}{2}\bar{Z}^{(k)} - (I - Z^{(k)})(I + \frac{1}{2}\bar{Z}^{(k)})] \\ &= (I - \frac{1}{2}\bar{Z}^{(k)})^{-1}[I - \frac{1}{2}\bar{Z}^{(k)} - I - \frac{1}{2}\bar{Z}^{(k)} + Z^{(k)} + \frac{1}{2}Z^{(k)}\bar{Z}^{(k)}] \\ &= (I - \frac{1}{2}\bar{Z}^{(k)})^{-1}[Z^{(k)} - \bar{Z}^{(k)} + \frac{1}{2}Z^{(k)}\bar{Z}^{(k)}] \\ &= (I - \frac{1}{2}\bar{Z}^{(k)})^{-1}[(Z^{(k)} - \bar{Z}^{(k)}) + \frac{1}{2}Z^{(k)}(\bar{Z}^{(k)} - Z^{(k)} + Z^{(k)})]. \end{aligned}$$

It then follows that

$$\begin{aligned} \|Z^{(k+1)}\| &\leq \|(I - \frac{1}{2}\bar{Z}^{(k)})^{-1}\|(\|Z^{(k)} - \bar{Z}^{(k)}\| + \frac{1}{2}\|Z^{(k)}\|^2 + \frac{1}{2}\|\bar{Z}^{(k)} - Z^{(k)}\|\|Z^{(k)}\|) \\ &\leq \frac{128}{223}\|Z^{(k)}\| + \frac{1}{32}\|Z^{(k)}\| + \frac{32}{223}\|Z^{(k)}\| \simeq \frac{1}{2}\|Z^{(k)}\|, \end{aligned}$$

because of (3.11) and (3.13). Now (3.7) combined with Lemma 3 shows the convergence of $\{c^{(k)}\}$, and this completes the proof. \square

Now we show the convergence of Algorithm 3. We replace our assumption by the following.

1. There exists c^* such that the matrix pencil $(A(c^*), B(c^*))$ has the given eigenvalues $\lambda_1^* = \dots = \lambda_t^* < \lambda_{t+1}^* < \dots < \lambda_{n-s}^*$.
2. The matrix $K(c^*)$, defined by (2.23) and (2.24) using any B -orthonormal set of eigenvectors of $(A(c^*), B(c^*))$, is nonsingular.

Theorem 3. *There exists θ such that, if $\|Z^{(k)}\| \leq \theta$ for some k , then the sequence of norms of the matrix $Z^{(k+1)}$ of Algorithm 3 converges to zero, and the iterates $\{c^{(k)}\}$ of Algorithm 3 converge quadratically to c^* .*

Proof. First, we bound $\|c^{(k+1)} - c^*\|$ in a similar way as (3.7). The only difference is that the new iterate $c^{(k+1)}$ is defined by (2.23) and (2.24). So, using the first $n-s$ diagonal equations of (3.6), plus the equations corresponding to $1 \leq i < j \leq t$, and the definition of matrix K , we have

$$\|c^* - c^{(k+1)}\| \leq 2\sqrt{n}\|Z^{(k)}\|^2\|\Lambda^*\|\|K^{(k)}\|^{-1}. \quad (3.14)$$

For the second part we need to consider the unspecified eigenvalues. So by using the off-diagonal elements in (3.6) and the new eigenvalues $\{\bar{\lambda}_i^*\}$, we have

$$|\bar{Z}_{ij}^{(k)} - Z_{ij}^{(k)}| \leq \frac{|q_j^{(k)T} (\sum_{l=1}^n (c_l^* - c_l^{(k+1)})(A_l - \bar{\lambda}_j^* B_l) q_i^{(k)})| + 2|\bar{\lambda}_j^*| \|Z^{(k)}\|^2}{|\bar{\lambda}_i^* - \bar{\lambda}_j^*|}, \quad i < j, j > t. \quad (3.15)$$

As explained before, by introducing a tolerance parameter ϵ in Step 7 of Algorithm 3, we ensure that the unspecified eigenvalues are distinct from the given eigenvalues. Also, the diagonal elements of $\bar{Z}^{(k)} - Z^{(k)}$ have a similar bound as in (3.15). Finally, since $Z_{ij} = 0, 1 \leq i \leq j \leq t$, as in (3.11), we have

$$\|\bar{Z}^{(k)} - Z^{(k)}\| \leq \frac{2\sqrt{n^2 - t^2} \|\Lambda^*\| \|Z^{(k)}\|^2 (1 + \rho)}{\delta(\bar{\lambda}^*) (1 - \rho(2 + \|Z^{(k)}\|) \|Z^{(k)}\|)}. \quad (3.16)$$

where $\delta(\bar{\lambda}^*) = \min_{i \neq j} |\bar{\lambda}_i^* - \bar{\lambda}_j^*|$. Now suppose that $\theta = \frac{\delta(\bar{\lambda}^*)}{8\sqrt{n^2 - t^2} \|\Lambda^*\| (1 + \rho)}$, and due to this assumption, for $\rho \geq 1, 1 \leq t \leq n$ and $n \geq 2$, we see $\|Z^{(k)}\| \leq \theta \leq \frac{1}{8\sqrt{3}} \leq \frac{1}{8}$.

The rest of the proof is the same as the proof in Theorem 2 and we have

$$\|Z^{(k+1)}\| \leq \|(I - \frac{1}{2}\bar{Z}^{(k)})^{-1} (\|Z^{(k)} - \bar{Z}^{(k)}\| + \frac{1}{2}\|Z^{(k)}\|^2 + \frac{1}{2}\|\bar{Z}^{(k)} - Z^{(k)}\| \|Z^{(k)}\|). \quad (3.17)$$

Hence, in a similar way as the proof in Theorem 2, we obtain

$$\|Z^{(k+1)}\| \leq \frac{32}{49} \|Z^{(k)}\| + \frac{1}{8} \|Z^{(k)}\| + \frac{32}{89} \|Z^{(k)}\|.$$

Also, the assumption of the nonsingularity of matrix $K(c^{(k)})$ combined with (3.14) shows the quadratic convergence of $c^{(k)}$ and the proof is complete. \square

4. Numerical experiments

In this section, we use four examples to examine the convergence of Algorithm 2. In addition, we present numerical tests of Algorithm 3 on two parameterized generalized inverse eigenvalue problems with multiple eigenvalues, and show the effectiveness of this algorithm for iteratively computing a solution of Problem 3. Also, in some examples, we compare the performance and numerical accuracy of the new methods with some other available algorithms.

It is noteworthy that in our tests, the iterations were terminated when the current iterate satisfies $\max\{\|Q^{(k+1)T} A(c^{(k+1)}) Q^{(k+1)} - \Lambda^*\|, \|Q^{(k+1)T} B(c^{(k+1)}) Q^{(k+1)} - I\|\} \leq 10^{-12}$ for Algorithm 2, and in Algorithm 3 when the residual of Step 4 was less than 10^{-8} . All our experiments were carried out in MATLAB. In all the numerical examples, we compute the initial matrix $Q^{(0)}$ as the eigenvector matrix of the pencil $(A(c^0), B(c^0))$. In some examples where the eigenvalues $\{\lambda_i^*\}$ are unknown, we first produce c^* with entries randomly selected, then we compute the eigenvalues of $(A(c^*), B(c^*))$. Examples are presented in the following sub-sections to assess the accuracy and performance.

4.1. Accuracy evaluation

In this sub-section, we illustrate the convergence performance of the proposed methods on some examples. For all examples, we report our experimental results in tables where the columns labeled as E_1 and E_2 show the values of residues $\|Q^{(k+1)T} A(c^{(k+1)}) Q^{(k+1)} - \Lambda^*\|$ and $\|Q^{(k+1)T} B(c^{(k+1)}) Q^{(k+1)} - I\|$, respectively, and k represents the iteration number.

Table 1: Numerical results for Example 1 by using Algorithm 2.

k	E_1	E_2	$\ Z^{(k)}\ $	$\ c^{(k+1)} - c^{(k)}\ $
0	2.107e-02	7.421e-04	1.364e+00	2.424e-01
1	3.794e-03	1.111e-04	1.542e+00	1.021e-01
2	2.351e-05	2.259e-09	7.380e-02	6.904e-03
3	1.628e-09	5.283e-16	1.125e-03	8.316e-05
4	4.709e-16	2.778e-16	2.072e-08	2.119e-09

Example 1. For the first example, we consider a typical problem in [29, Example 1]. We have a parameterized generalized inverse eigenvalue problem in which $n = 5$,

$$A_0 = \text{diag}(9, 11, 10, 8, 14), \quad B_0 = \text{diag}(11, 13, 15, 11, 10), \quad A_1 = B_1 = I,$$

$$A_2 = \begin{bmatrix} 0 & 2 & 0 & 0 & 0 \\ 2 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 \end{bmatrix},$$

$$A_3 = \begin{bmatrix} 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 3 & 0 & 0 & 0 & -1 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{bmatrix}, \quad B_3 = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix},$$

$$A_4 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}, \quad B_4 = \begin{bmatrix} 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix},$$

$$A_5 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} = B_5,$$

$$A(c) = A_0 + \sum_{i=1}^5 c_i A_i, \quad B(c) = B_0 + \sum_{i=1}^5 c_i B_i.$$

The eigenvalues are defined to be

$$\lambda^* = [0.43278721102, 0.66366274839, 0.94385900467, 1.10928454002, 1.49235323254]^T.$$

Algorithm 2 is used to find the unknown parameters vector c . This algorithm running with the starting vector $c^{(0)} = [1.1, 1.2, 1.3, 1.4, 1.5]^T$ converges to the solution

$$c^* = [1, 1, 1, 1, 1]^T.$$

The numerical results for Algorithm 2 are displayed in Table 1.

Table 2: Numerical results for Example 2 by using Algorithm 2.

n	k	E_1	E_2	$\ Z^{(k)}\ $	$\ c^{(k+1)} - c^{(k)}\ $
100	0	2.107e-02	7.421e-04	1.364e+00	2.424e-01
	1	2.067e-03	3.056e-04	5.787e-02	3.403e-01
	2	8.524e-06	1.117e-07	76.613e-04	2.033e-02
	3	5.325e-13	2.840e-13	2.581e-07	8.625e-06
	4	1.730e-13	4.077e-13	8.557e-11	7.541e-12
200	0	2.850e-03	3.051e-04	7.104e-0	3.689e-01
	1	2.807e-05	1.444e-07	8.862e-04	7.104e-01
	2	2.087e-12	1.510e-12	4.087e-07	1.856e-05
	3	3.814e-13	1.361e-12	4.739e-10	4.519e-11

Example 2 [14]. In this example, we construct a matrix $A(c)$ as (1.1), using Toeplitz matrices $\{A_i\}$ in the form

$$A_0 = 0, \quad A_1 = I, \quad A_2 = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 1 & 0 & 1 & \ddots & \vdots \\ 0 & 1 & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & 0 & 1 \\ 0 & \dots & 0 & 1 & 0 \end{bmatrix}, \dots, A_n = \begin{bmatrix} 0 & 0 & \dots & 0 & 1 \\ 0 & \ddots & \ddots & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & \ddots & \ddots & 0 \\ 1 & 0 & \dots & 0 & 0 \end{bmatrix}.$$

In addition, to construct a matrix $B(c)$ as (1.3), we use the set of matrices $\{B_i\}$ given by

$$B_0 = 0, \quad B_k = \{b_{ij}^k\}, \quad b_{ij}^k = \begin{cases} 1 & \text{if } i = j = k, \\ 0 & \text{otherwise.} \end{cases}$$

In Table 2, for $n = 100$ and 200 , the convergence histories of Algorithm 2 are illustrated.

Example 3. We consider an example obtained from a physical application. This is a special case of the famous on-line system of masses and springs [3, 8]. Suppose that m_i and k_i (for $i = 1, 2, \dots, n$) are the mass of the i th particle and the stiffness of the i th spring, respectively, and that the springs satisfy Hooke's law. Then the motion equations are given by

$$m_i \frac{d^2 u_i}{dt^2} = \begin{cases} -k_i u_i + k_{i+1}(u_{i+1} - u_i), & i = 1, \\ -k_i(u_i - u_{i-1}) + k_{i+1}(u_{i+1} - u_i), & i = 2, 3, \dots, n-1, \\ -k_i(u_i - u_{i-1}), & i = n. \end{cases} \quad (4.1)$$

We can write the matrix form of equations (4.1) as

$$M \frac{d^2 u}{dt^2} = -K u,$$

where

$$K = \begin{bmatrix} k_1 + k_2 & -k_2 & 0 & \dots & 0 & 0 \\ -k_2 & k_2 + k_3 & -k_3 & & & 0 \\ 0 & -k_3 & k_3 + k_4 & \ddots & & 0 \\ \vdots & & \ddots & \ddots & & \vdots \\ 0 & & & & & -k_n \\ 0 & 0 & \dots & -k_n & k_n & \end{bmatrix}, \quad M = \text{diag}(m_1, m_2, \dots, m_n).$$

M and K are called the mass and stiffness matrices, respectively. A fundamental solution $u(t) = e^{i\omega t} x$ result in the symmetric eigenvalue problem $Kx = \lambda Mx$, where $\lambda = \omega^2$. For finding the stiffness parameters

Table 3: Numerical results for Example 3 by using Algorithm 2.

n	k	E_1	E_2	$\ Z^{(k)}\ $	$\ c^{(k+1)} - c^{(k)}\ $
100	0	4.513e-03	1.167e-14	6.630e-02	1.247e+00
	1	3.283e-03	2.431e-14	3.074e-02	9.635e-01
	2	1.009e-06	3.601e-14	5.257e-03	1.788e-01
	3	1.707e-09	1.386e-14	2.263e-04	7.667e-03
	4	6.337e-15	1.638e-14	4.130e-07	1.393e-05
5	6.040e-15	1.358e-14	6.882e-12	2.819e-10	
200	0	4.896e-05	3.327e-14	2.908e-02	1.596e+00
	1	7.548e-06	4.714e-14	1.141e-02	6.279e-01
	2	2.209e-07	4.503e-14	1.962e-03	2.936e-01
	3	7.343e-09	3.970e-14	3.577e-04	1.955e-02
	4	9.314e-12	3.770e-14	1.274e-05	6.964e-04
5	2.618e-14	1.554e-14	1.512e-08	8.261e-07	

k_i corresponding to given natural frequencies $\omega_1 < \omega_2 < \dots < \omega_n$, we have an inverse problem. It is easily seen that this inverse problem can be modeled as Problem 2, using the notation,

$$K = K_0 + \sum_{i=1}^n k_i K_i,$$

where

$$\begin{aligned} K_0 &= 0, & K_1 &= e_1 e_1^T, \\ K_i &= (e_{i-1} - e_i)(e_{i-1} - e_i)^T, & i &= 2, 3, \dots, n, \\ M_0 &= M, & M_i &= 0, \quad i = 1, 2, \dots, n, \end{aligned}$$

where e_k is the k th column of the identity matrix. For simplicity, here we assume $M = I$.

We consider the two problem sizes $n = 100$ and 200 . By using the initial guess $c^{(0)} = c^* + \zeta$, where $c^* = [1, 2, 3, \dots, n-1, n]^T$ and $\zeta = 0.5$, Algorithm 2 converges. The residual values for this problem are given in Table 3.

Example 4. In this example, we consider a truss problem and present some computational results to demonstrate the advantages of Problem 2 in structural design. We consider an n -bar truss problem [24], with the following parameters: Young's modulus $E = 6.95 \times 10^{10} N/m^2$, weight density $P = 2650 kg/m^3$, acceleration of gravity $g = 9.81 m/s^2$, non-structural mass at all nodes $m_0 = 425 kg$, and length $l = 10m$ of horizontal and vertical bars. In this problem, the design variables are the areas of cross sections of the bars. Using c_i to denote the area of the cross section of the i th bar, the stiffness and mass matrices of the structure can be written respectively as

$$A(c) = A_0 + \sum_{i=1}^8 c_i A_i, \quad B(c) = B_0 + \sum_{i=1}^8 c_i B_i,$$

where $A_0 = 0$, $B_0 = m_0 I$ is the constant lumped mass matrix, and A_i, B_i ($i = 1, 2, \dots, 8$) are $n \times n$ symmetric matrices.

Assuming that $c^* = [1, 1, \dots, 1]^T$ and using the starting vector $c^{(0)} = c^* + \zeta$, where each element of ζ is 0.5, the algorithm converges. In Table 4, for $n = 100$ and 200 , the convergence histories are illustrated.

Table 4: Numerical results for Example 4 by using Algorithm 2.

n	k	E_1	E_2	$\ Z^{(k)}\ $	$\ c^{(k+1)} - c^{(k)}\ $
100	0	2.108e+04	6.874e-02	1.665e-01	5.000e+00
	1	7.541e+02	2.466e-03	3.436e-02	3.209e-04
	2	9.238e-01	3.041e-06	1.233e-03	3.550e-05
	3	1.384e-06	4.705e-12	1.530e-06	1.487e-07
	4	4.670e-09	3.998e-14	2.565e-11	1.399e-11
200	0	2.108e+04	6.877e-02	1.675e-01	7.071e+00
	1	7.546e+02	2.470e-03	3.592e-02	9.582e-04
	2	9.291e-01	3.073e-06	1.952e-03	1.125e-04
	3	1.405e-06	4.804e-12	8.047e-06	2.833e-07
	4	3.412e-08	2.658e-13	8.446e-11	2.674e-11

Example 5 [29]. This example is a Problem 3 with multiple eigenvalues and $n = 6$. Define

$$A_0 = \begin{bmatrix} 216 & 889.2 & -135 & -245.4 & 141.6 & 858.12 \\ 889.2 & 66.3 & -483.75 & -820.39 & 413.12 & 1598.29 \\ -135 & -483.75 & 204.375 & -2.425 & -131.5 & -541.325 \\ -245.4 & -820.39 & -2.425 & 158.495 & -367.365 & 693.021 \\ 141.6 & 413.12 & -131.5 & -367.365 & 209.035 & 154.057 \\ 858.12 & 1598.29 & -541.325 & 693.021 & 154.057 & 316.63 \end{bmatrix}, \quad B_0 = I,$$

$B_1 = \text{diag}(43, 2222995816, 43.5245534248, 43.2978630424, 43.6484775005, 43.3099531961, 43.6635901927)$,

and matrices A_k and B_k are determined as

$$A_k = u_k u_k^T, \quad k = 1, 2, \dots, 6,$$

$$B_k = \sum_{j=k}^6 V_{k-1,j} (e_{k-1} e_j^T + e_j e_{k-1}^T), \quad k = 1, 2, \dots, 6,$$

where

$$u_1 = \begin{bmatrix} 12 \\ -1 \\ 0.5 \\ -0.4 \\ 0.2 \\ 0.1 \end{bmatrix}, \quad u_2 = \begin{bmatrix} 12 \\ -1 \\ 0.5 \\ -0.4 \\ 0.2 \\ 0.1 \end{bmatrix}, \quad u_3 = \begin{bmatrix} 0 \\ 0 \\ 12 \\ -1 \\ 0.5 \\ -0.4 \end{bmatrix},$$

$$u_4 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 12 \\ 0 \\ 0.5 \end{bmatrix}, \quad u_5 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 12 \\ 0.1 \end{bmatrix}, \quad u_6 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 12 \end{bmatrix},$$

$$V = \begin{bmatrix} 0 & -0.8467500959 & 0.4233750479 & -0.3387000384 & 0.1693500192 & -0.0846750096 \\ -0.8467500959 & 0 & -0.2246513095 & 2.8755367617 & -0.0898605238 & 1.1232565475 \\ 0.4233750479 & -0.2246513095 & 0 & -0.9004508446 & 0.4502254223 & -0.3579661145 \\ -0.3387000384 & 2.8755367617 & -0.9004508446 & 0 & -0.6097095231 & 6.8750004851 \\ 0.1693500192 & -0.0898605238 & 0.4502254223 & -0.6097095231 & 0 & -0.6911965633 \\ -0.0846750096 & 1.1232565475 & -0.3579661145 & 6.8750004851 & -0.6911965633 & 0 \end{bmatrix}.$$

Table 5: Numerical results for Example 5 by using Algorithm 3.

k	E_1	E_2	$\ Z^{(k)}\ $	$\ c^{(k+1)} - c^{(k)}\ $
0	2.317e-01	1.491e-01	3.561e-01	2.239e+00
1	1.739e-02	4.661e-03	1.152e-01	3.855e-01
2	8.811e-05	3.359e-05	5.407e-03	5.543e-02
3	5.962e-09	5.916e-10	7.640e-05	1.468e-04
4	1.656e-15	1.067e-15	5.850e-09	1.958e-08

Also, let $c^* = [3.3635, 14.1327, 2.2459, 13.5385, 0.9773, 17.8286]$, then the pencil $(A(c^*), B(c^*))$ has the eigenvalues

$$\lambda_1^* = \lambda_2^* = \lambda_3^* = 1.5, \lambda_4^* = 3.2124, \lambda_5^* = 18.9881, \lambda_6^* = 33.3574.$$

Algorithm 2, on the assumption that the starting vector is $c^{(0)} = [3, 15, 3, 15, 1, 18]^T$ and $t = s = 3$, converges to the solution

$$c^* = [3.3635, 14.1327, 2.2459, 13.5385, 0.9773, 17.8286]^T.$$

In Table 5, we report the obtained results.

Example 6. Let $n = 8$, $A_0 = 0$, $B_0 = \text{diag}(99, 103, 100, 105, 101, 107, 101, 108)$ and $B_1 = I$. In addition, let A and B be two symmetric matrices of the form

$$A = \begin{bmatrix} 7400 & 11820 & -12060 & 1850 & -3020 & 460 & 710 & -2326 \\ 11820 & 27176 & -29158 & 7100 & -6116 & 2928 & 1878 & -4992.8 \\ -12060 & -29158 & 32914 & -8005 & 6318 & -3854 & -1759 & 5681.4 \\ 1850 & 7100 & -8005 & 2950 & -1274 & 1466 & 827 & -1106 \\ -3020 & -6116 & 6318 & -1274 & 2144 & -166 & 6 & 1334 \\ 460 & 2928 & -3854 & 1466 & -166 & 1523 & 593 & -351.2 \\ 710 & 1878 & -1759 & 827 & 6 & 593 & 990 & 12.8 \\ -2326 & -4992.8 & 5681.4 & -1106 & 1334 & -351.2 & 12.8 & 1408.76 \end{bmatrix},$$

$$B = \begin{bmatrix} 0 & -20 & -10 & -10 & -10 & 10 & 0 & -10 \\ -20 & 0 & -20 & 12 & 2 & -12 & 0 & 2 \\ -10 & -20 & 0 & 21 & 9 & -9 & -10 & 9 \\ -10 & 12 & 21 & 0 & -1 & 10 & 2 & -11 \\ -10 & 2 & 9 & -1 & 0 & -22 & 9 & -8 \\ 10 & -12 & -9 & 10 & -22 & 0 & -1 & 9 \\ 0 & 0 & -10 & 2 & 9 & -1 & 0 & -22 \\ -10 & 2 & 9 & -11 & -8 & 9 & -22 & 0 \end{bmatrix}.$$

Now construct the matrices A_k and B_k from A and B as

$$A_k = \sum_{j=1}^{k-1} a_{kj}(e_k e_j^T + e_j e_k^T) + a_{kk} e_k e_k^T \quad k = 1, 2, \dots, 8,$$

$$B_k = \sum_{j=1}^{k-1} b_{k-1,j}(e_{k-1} e_j^T + e_j e_{k-1}^T), \quad k = 2, 3, \dots, 8,$$

where e_k is the k th column of the identity matrix. This example was presented in [29].

We consider $c^* = [1, 1, 1, 1, 1, 1, 1, 1]^T$, then the generalized eigenvalue problem for $(A(c^*), B(c^*))$ has one multiple eigenvalue of multiplicity 3,

$$\lambda_1^* = \lambda_2^* = \lambda_3^* = 2, \lambda_4^* = 3, \lambda_5^* = 10, \lambda_6^* = 16.9485, \lambda_7^* = 34.9037, \lambda_8^* = 586.4204.$$

Table 6: Numerical results for Example 6 by using Algorithm 3.

k	E_1	E_2	$\ Z^{(k)}\ $	$\ c^{(k+1)} - c^{(k)}\ $
0	1.273e-01	5.357e-03	1.794e-01	3.429e-01
1	4.328e-03	1.518e-04	2.743e-02	6.268e-02
2	1.067e-05	9.780e-08	1.278e-03	1.639e-03
3	2.722e-11	6.364e-13	2.824e-06	3.265e-06
4	3.209e-14	1.192e-14	5.655e-12	6.117e-12

Table 7: Comparison between the proposed algorithm and Dai *et al.* on Example 2.

n	Algorithm 2		Algorithm 4.1 [7]	
	$\ c^{(k)} - c^*\ $	CPU	$\ c^{(k)} - c^*\ $	CPU
40	3.448e-13	0.085	3.991e-08	1.274
80	2.097e-12	0.425	4.900e-08	18.515
120	9.590e-12	0.982	1.234e-10	99.696
160	2.549e-11	1.315	3.074e-11	379.765
200	4.519e-11	2.066	4.501e-11	984.808
240	1.018e-10	3.261	-	-
400	5.057e-10	9.296	-	-

Therefore, according to the formulation of Problem 3, we have $t = 3$, $s = 3$ and $n - s = 5$. So, the target eigenvalues were selected as

$$\lambda^* = [2, 2, 2, 3, 10]^T.$$

With the starting vector

$$c^{(0)} = [0.98, 0.97, 0.98, 0.99, 1, 1.01, 1.02, 1.03]^T,$$

after 5 iterations, the locally unique solution found by the modified algorithm is

$$c^* = [0.9456, 0.9411, 0.9424, 0.9374, 0.9399, 0.9358, 0.9009, 0.9285]^T.$$

We show the results in Table 6.

From the above six examples, we can observe that Algorithms 2 and 3 are feasible for solving Problems 2 and 3, respectively.

4.2. Performance evaluation

In this sub-section, we compare the performance and numerical accuracy of Algorithm 2 and Algorithm 4.1 in [7], by testing Examples 2, 3 and 4. The numerical results and CPU times for these cases can be seen in Tables 7, 8 and 9, respectively. It should be noted that in Examples 2 and 4, for large dimensions, Algorithm 4.1 requires a lot of time. But in Example 3, our implementation showed that the sequence generated by algorithm 4.1 fails to converge. Our numerical results prove that Algorithm 2 is more effective than Algorithm 4.1 in [7] in terms of the computing time.

Table 8: Comparison between the proposed algorithm and Dai *et al.* on Example 3.

n	Algorithm 2		Algorithm 4.1 [7]	
	$\ c^{(k)} - c^*\ $	CPU	$\ c^{(k)} - c^*\ $	CPU
40	3.910e-11	0.0624	8.061e-11	1.087
80	3.992e-10	1.107	-	-
120	7.752e-10	7.889	-	-
160	1.370e-09	16.321	-	-
200	4.094e-09	51.8698	-	-
240	6.227e-09	70.633	-	-

Table 9: Comparison between the proposed algorithm and Dai *et al.* on Example 4.

n	Algorithm 2		Algorithm 4.1 [7]	
	$\ c^{(k)} - c^*\ $	CPU	$\ c^{(k)} - c^*\ $	CPU
40	3.448e-12	0.089	6.168e-11	0.918
80	2.097e-12	0.414	2.978e-12	4.820
120	9.590e-12	1.462	2.113e-12	20.176
160	4.519e-11	2.305	1.885e-11	68.595
200	4.519e-11	2.066	1.407e-11	165.678
240	2.800e-11	3.261	9.773e-11	355.977
400	1.669e-10	11.470	-	-

5. Conclusions

In this paper, we first provided an algorithm based on the Cayley transform for solving a type of parameterized generalized inverse eigenvalue problem with distinct eigenvalues. When the given eigenvalues are multiple, we have explained how to express the problem so that it is not overdetermined, and then we have presented an algorithm for these cases. We have analyzed the convergence for both the distinct and the multiple eigenvalue cases. Numerical examples have been given to show the agreement with the theoretical results. Also, we showed that they are applicable to large problems. In particular, our methods are much faster with respect to the methods proposed in [7], and can address much larger problem sizes.

Acknowledgements. This research was developed during a visit of the first author to Universitat Politècnica de València. This visit was funded by Ministry of Science, Research and Technology of Iran. The second author was partially supported by the Spanish Agencia Estatal de Investigación (AEI) under grant TIN2016-75985-P, which includes European Commission ERDF funds. We thank Carmen Campos for useful comments on an initial draft of the paper.

References

- [1] M. Chu, G. Golub, Inverse Eigenvalue Problems, Algorithms and Applications, Oxford University Press, Oxford, 2005.
- [2] M. Hajarian, H. Abbas, Least squares solutions of quadratic inverse eigenvalue problem with partially bisymmetric matrices under prescribed submatrix constraints, *Comput. Math. Appl.* 76(6) (2018) 1458–1475.
- [3] M. Chu, G. Golub, Structured inverse eigenvalue problems, *Acta Numerica* 11 (2002) 1–71.
- [4] K. Ghanbari, F. Parvizpour, Generalized inverse eigenvalue problem with mixed eigendata, *Linear Algebra Appl.* 437(8) (2012) 2056–2063.
- [5] Y. X. Yuan, H. Dai, A generalized inverse eigenvalue problem in structural dynamic model updating, *J. Comput. Appl. Math.* 226(1) (2009) 42–49.
- [6] S. F. Yuan, Q. W. Wang, Z. P. Xiong, Linear parameterized inverse eigenvalue problem of bisymmetric matrices, *Linear Algebra Appl.* 439 (7) (2013) 1990–2007.
- [7] H. Dai, Z. Z. Bai, Y. Wei, On the solvability condition and numerical algorithm for the parameterized generalized inverse eigenvalue problem, *SIAM J. Matrix Anal. Appl.* 36 (2) (2015) 707–726. doi:10.1137/140972494.
- [8] G. M. Gladwell, Inverse problems in vibration, *Appl. Mech. Rev.* 39 (7) (1986) 1013–1018.
- [9] S. Friedland, J. Nocedal, M. L. Overton, The formulation and analysis of numerical methods for inverse eigenvalue problems, *SIAM J. Numer. Anal.* 24 (3) (1987) 634–667. doi:10.1137/0724043.
- [10] R. H. Chan, H. L. Chung, S. F. Xu, The inexact Newton-like method for inverse eigenvalue problem, *BIT* 43 (1) (2003) 7–20. doi:10.1023/a:1023611931016.
- [11] Z. J. Bai, R. H. Chan, B. Morini, An inexact Cayley transform method for inverse eigenvalue problems, *Inverse Problems* 20 (5) (2004) 1675–1689. doi:10.1088/0266-5611/20/5/022.
- [12] W. P. Shen, C. Li, X. Q. Jin, A Ulm-like method for inverse eigenvalue problems, *App. Numer. Math.* 61 (3) (2011) 356–367. doi:10.1016/j.apnum.2010.11.001.
- [13] W. Shen, C. Li, An Ulm-like Cayley transform method for inverse eigenvalue problems, *Taiwan. J. Math.* 16 (1) (2012) 367–386. doi:10.11650/twjm/1500406546.
- [14] K. Aishima, A quadratically convergent algorithm based on matrix equations for inverse eigenvalue problems, *Linear Algebra Appl.* 542 (2018) 310–333. doi:10.1016/j.laa.2017.05.019.
- [15] W. P. Shen, C. Li, X. Q. Jin, An inexact Cayley transform method for inverse eigenvalue problems with multiple eigenvalues, *Inverse Problems* 31 (8) (2015) 085007. doi:10.1088/0266-5611/31/8/085007.
- [16] W. Shen, C. Li, X. Q. Jin, An Ulm-like Cayley transform method for inverse eigenvalue problems with multiple eigenvalues, *Numer. Math. Theor. Meth. Appl.* 9 (04) (2016) 664–685. doi:10.4208/nmtma.2016.y15030.

- [17] K. Aishima, A quadratically convergent algorithm for inverse eigenvalue problems with multiple eigenvalues, *Linear Algebra Appl.* 549 (2018) 30–52. doi:10.1016/j.laa.2018.03.022.
- [18] L. Li, Sufficient conditions for the solvability of algebraic inverse eigenvalue problems, *Linear Algebra Appl.* 221 (1995) 117–129.
- [19] S. Xu, On the sufficient conditions for the solvability of algebraic inverse eigenvalue problems, *J. Comput. Math* 10 (1992) 17–80.
- [20] F. Biegler-König, Sufficient conditions for the solubility of inverse eigenvalue problems, *Linear Algebra Appl.* 40 (1981) 89–100.
- [21] J. Alexander, The additive inverse eigenvalue problem and topological degree, *Proc. Amer. Math. Soc* 70 (1978) 5–7.
- [22] C. I. Byrnes, X. Wang, The additive inverse eigenvalue problem for Lie perturbations, *SIAM J. Matrix Anal. Appl.* 14(1) (1993) 113–117.
- [23] Z. Wang, S. Vong, A Gauss–Newton-like method for inverse eigenvalue problems, *Int. J. Comput. Math.* 90 (7) (2013) 1435–1447. doi:10.1080/00207160.2012.750721.
- [24] K. Joseph, Inverse eigenvalue problem in structural design, *Numer. Linear Algebra Appl.* 30(12) (1992) 2890–2896.
- [25] J. Jiang, H. Dai, Y. Yuang, A symmetric generalized inverse eigenvalue problem in structural dynamics model updating, *Linear Algebra Appl.* 439(5) (2013) 1350–1363.
- [26] S. J. Cox, M. Embree, J. M. Hokanson, One can hear the composition of a string: experiments with an inverse eigenvalue problem, *SIAM Review* 54(1) (2012) 157–178.
- [27] S. Mulaik, *Fundamentals of common factor analysis*, The Wiley Handbook of Psychometric Testing: A Multidisciplinary Reference on Survey, Scale and Test Development (2018) 209–251.
- [28] X. Ji, On matrix inverse eigenvalue problems, *Inverse problems* 14(2) (1998) 275.
- [29] H. Dai, P. Lancaster, Newton’s method for a generalized inverse eigenvalue problem, *Numer. Linear Algebra Appl.* 4 (1) (1997) 1–21.
- [30] L. Shu, B. Wang, J. Hu, Homotopy solution of the inverse generalized eigenvalue problems in structural dynamics, *Appl. Math. Mech.* 25(5) (2004) 580–586.
- [31] H. Dai, An algorithm for symmetric generalized inverse eigenvalue problems, *Linear Algebra Appl.* 296 (1999) 79–98.
- [32] P. Lancaster, Algorithms for lambda-matrices, *Numer. Math.* 6(1) (1964) 388–394.
- [33] F. Biegler-König, A Newton iteration process for inverse eigenvalue problems, *Numer. Math.* 37(3) (1981) 349–354.
- [34] S. Qi, D. Hua, Numerical methods for solving generalized inverse eigenvalue problem, *Journal of Nanjing University of Aeronautics and Astronautics* 26 (1999) 1–94.
- [35] B. N. Parlett, *The symmetric eigenvalue problem*, Vol. 20, SIAM, 1998.
- [36] N. J. Higham, *Functions of Matrices: Theory and Computation*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2008.