East Asian Journal on Applied Mathematics
doi: 10.4208/eajam.100916.061117a

Vol. **8**, No. 1, pp. 139-150
February 2018

A Modified Newton Method for Nonlinear Eigenvalue Problems

Xiao-Ping Chen[1,2] and Hua Dai[1,*]

[1] *Department of Mathematics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China.*
[2] *Department of Mathematics, Taizhou University, Taizhou 225300, China.*

Received 10 September 2016; Accepted (in revised version) 6 November 2017.

Abstract. A modification to the Newton method for nonlinear eigenvalue problems is proposed and locally quadratic convergence of this algorithm is established. Numerical examples show the efficiency of the method and reduced computational cost.

AMS subject classifications: 65F15

Key words: Nonlinear eigenvalue problem, smallest singular value, Newton method, quadratic convergence.

1. Introduction

In this paper we consider the nonlinear eigenvalue problem (NEP)

$$T(\lambda)x = 0, \tag{1.1}$$

where $T(\lambda)$ is an analytic $n \times n$ matrix-valued function of complex variable λ. The problem (1.1) includes the classical eigenvalue problem $(T(\lambda) = \lambda I - A)$, the generalised eigenvalue problem $(T(\lambda) = \lambda A - B)$, the quadratic eigenvalue problem $(T(\lambda) = \lambda^2 A + \lambda B + C)$ — [16, 27], the polynomial eigenvalue problem $(T(\lambda) = \lambda^m A_m + \lambda^{m-1} A_{m-1} + \cdots + \lambda A_1 + A_0)$ — [6, 16] and the delay eigenvalue problem $(T(\lambda) = \lambda I - A - \sum_{i=1}^{m} A_i e^{-\tau_i \lambda})$ — [10,11]. Nonlinear eigenvalue problems arise in various applications, including nonlinear ordinary differential equations [26], acoustic surface waves [32], photonic band structures [24], vibration of viscoelastic structures [1,5] and fluid-solid structures [3,30], simulation of quantum dots [8,31], the stability of time-delay systems [20] and so on. For more information about possible applications of the NEP (1.1) and numerical methods for its solution the reader can consult Ref. [19].

It is well known that λ is an eigenvalue of the problem (1.1) if and only if it satisfies the characteristic equation

$$\det\big(T(\lambda)\big) = 0. \tag{1.2}$$

[*] Corresponding author. *Email addresses:* cxpnuaa@163.com (X.-P. Chen), hdai@nuaa.edu.cn (H. Dai)

The root of the equation (1.2) can be derived by the Newton method

$$\lambda_k = \lambda_{k-1} - \frac{1}{\mathrm{tr}\,(T(\lambda_{k-1})^{-1} T'(\lambda_{k-1}))}, \quad k = 1, 2, \cdots,$$

where $T'(\lambda_{k-1})$ is the derivative of $T(\lambda)$ at the point $\lambda = \lambda_{k-1}$ and $\mathrm{tr}(A)$ denotes the trace of the matrix A — cf. Ref. [13,16]. If an eigenvalue of the matrix T is known, the associated eigenvector can be constructed by the inverse iteration [21].

Let $T(\lambda)P(\lambda) = Q(\lambda)R(\lambda)$ be a QR decomposition with column pivoting of the matrix $T(\lambda)$ — i.e. $P(\lambda)$ and $Q(\lambda)$ are, respectively, permutation and unitary matrices, and the matrix $R(\lambda) = (r_{ij}(\lambda))$ is upper-triangular with the diagonal entries ordered as

$$\left|r_{11}(\lambda)\right| \geq \left|r_{22}(\lambda)\right| \geq \cdots \geq \left|r_{nn}(\lambda)\right|.$$

Kublanovskaya [15] used the Newton method for the equation $r_{nn}(\lambda) = 0$ to find an eigenvalue of the NEP (1.1). Jian and Singhal [9] improved the approach of [15] and established the quadratic convergence of the method. Li [17,18] presented sufficient conditions for a smooth QR decomposition and developed numerical methods for NEP (1.1). A similar method, but based on a smooth LU decomposition of $T(\lambda)$, has been studied in [4, 32]. Nevertheless, for large matrices, the above methods are not as efficient as required.

If $\sigma_{min}(T(\lambda))$ is the smallest singular value of the matrix $T(\lambda)$, then λ is an eigenvalue of the NEP (1.1) if and only if it satisfies the equation

$$\sigma_{min}\big(T(\lambda)\big) = 0.$$

Hence, the singular value decomposition (SVD) of the matrix $T(\lambda)$ can be used to determine the solution of the NEP (1.1) and such an approach, combined with the Newton method, has been employed by Guo *et al.* — cf. Ref. [7]. On the other hand, the necessity to compute SVD of the corresponding matrix at each iteration step presents additional challenges in the method implementation. Therefore, the inverse iteration method is used to approximate the smallest singular value and the associated left and right singular vectors of $T(\lambda)$.

The paper is organized as follows. In Section 2, we consider the solution of the NEP (1.1) by the Newton method based on singular value decomposition and propose a modification of this approach. Section 3 deals with the convergence of the method. In particular, it is shown that convergence is locally quadratic. Numerical examples are discussed in Section 4, and our concluding remarks are in Section 5.

2. Modified Newton Method for NEP

Let I denote the identity matrix, $\mathrm{diag}(a_1, a_2, \cdots, a_n)$ the diagonal matrix with the diagonal entries $a_1, a_2, \cdots, a_n$, $T^H(\lambda)$ the conjugate transpose to the matrix $T(\lambda)$, and $\|A\|$ the Euclidean norm of a matrix (a vector) A. Any matrix A can be written in the form $A = [\alpha_1, \alpha_2, \cdots, \alpha_n]$, with the column-vectors α_j. This notation is used in what follows.

Let us consider an approach to the NEP (1.1) based on the Newton method combined with singular value decomposition

$$T(\lambda)=U(\lambda)\Sigma(\lambda)V^H(\lambda)$$

of the matrix $T(\lambda)$. We recall that $\Sigma(\lambda)=\operatorname{diag}(\sigma_1(\lambda),\cdots,\sigma_n(\lambda))$ with $\sigma_1(\lambda)\geq\cdots\geq\sigma_n(\lambda)\geq 0$, and $U(\lambda)=[u_1(\lambda),\cdots,u_n(\lambda)]$, $V(\lambda)=[v_1(\lambda),\cdots,v_n(\lambda)]\in\mathbb{C}^{n\times n}$ are unitary matrices. Then

$$\sigma_{min}\big(T(\lambda)\big)=\sigma_n(\lambda)=u_n^H(\lambda)T(\lambda)v_n(\lambda), \tag{2.1}$$

and according to Refs. [2,12,33], the derivative of $\sigma_{min}(T(\lambda))$ is

$$\sigma'_{min}\big(T(\lambda)\big)=u_n^H(\lambda)T'(\lambda)v_n(\lambda).$$

The method proposed by Guo et al. [7] for solving the NEP (1.1) can be summarized as follows.

Algorithm 2.1. Starting with an initial approximation λ_0, for $k=1,2,\cdots$, make the following steps:

1. Derive the singular value decomposition $T(\lambda_{k-1})=U(\lambda_{k-1})\Sigma(\lambda_{k-1})V^H(\lambda_{k-1})$ of the matrix $T(\lambda_{k-1})$.
2. Use Newton method

$$\lambda_k=\lambda_{k-1}-\sigma_n(\lambda_{k-1})/\big[u_n^H(\lambda_{k-1})T'(\lambda_{k-1})v_n(\lambda_{k-1})\big]$$

   to obtain the next approximation λ_k.
3. Repeat the procedure till the required accuracy is achieved.

Algorithm 2.1 is a time-consuming procedure since it involves the SVD of the matrix $T(\lambda)$ at each iteration. However, the smallest singular value and the associated left and right singular vectors can be approximated by inverse iteration — cf. Refs. [21, 28, 29, 34]. More precisely, if $T=T(\lambda_k)$, $\sigma=\sigma_{min}(T(\lambda_k))$, and u and v are, respectively, the corresponding left and right singular vectors, the inverse iteration procedure considered by Xu [34] is:

Algorithm 2.2. Starting with an initial approximation u_0, for $k=1,2,\cdots$, make the following steps:

1. Find the solution of the linear system $Tw_k=u_{k-1}$.
2. Set $v_k:=w_k/\|w_k\|$.
3. Find the solution of the linear system $T^Hy_k=v_k$.
4. Set $u_k:=y_k/\|y_k\|$.
5. Repeat the procedure till the required accuracy is achieved.

Let u be the final left singular vector u_k obtained by Algorithm 2.2. If we derive w from the equation $Tw = u$, then the smallest singular value of the matrix T and the associated right singular vector can be approximated as follows $\sigma \approx 1/\|w\|$ and $v \approx w/\|w\|$.

Thus taking into account the fact that an approximation of the smallest singular value and the associated left and right singular vectors can be derived by the inverse iteration procedure, we propose the following modified Newton method for solving the NEP (1.1).

Algorithm 2.3. Starting with an initial approximation λ_0, make the following steps:

1. Compute the smallest singular value σ_0 and the corresponding left u_0 and right v_0 unit singular vectors of $T(\lambda_0)$.
2. For $k = 1, 2, \cdots$, until convergence do
    - 2.1. Use Newton method $\lambda_k = \lambda_{k-1} - \sigma_{k-1}/u_{k-1}^H T'(\lambda_{k-1})v_{k-1}$ to find the next approximation λ_k.
    - 2.2. Derive x_k from the linear system $T(\lambda_k)x_k = u_{k-1}$.
    - 2.3. Set $v_k := x_k/\|x_k\|$.
    - 2.4. Derive y_k from the linear system $T^H(\lambda_k)y_k = v_k$.
    - 2.5. Set $u_k := y_k/\|y_k\|$.
    - 2.6. Set $\sigma_k := u_k^H T(\lambda_k)v_k$.

In Algorithm 2.3, the linear systems $T(\lambda_k)x_k = u_{k-1}$ and $T^H(\lambda_k)y_k = v_k$ are to be solved at each iteration. For small and medium-sized matrices this can be done by a direct method and by a preconditioned GMRES method for large-scale ones — cf. Ref. [23].

3. Locally Quadratic Convergence of Modified Newton Method

Let us now analyse the convergence of Algorithm 2.3.

Lemma 3.1. *If $A(z)$ is an analytic $m \times n$ matrix-valued function in an open set $\mathscr{D} \subseteq \mathbb{C}$, then $A(z)$ has an analytic singular value decomposition on $\mathscr{D}$.*

Proof. The proof is similar to Theorem 1 in Ref. [2]. □

In what follows, we will use the notation $a_n = \mathscr{O}(b_n)$ for infinitesimals of the same order as n tends to ∞.

Lemma 3.2. *Let $T(\lambda)$ be an analytic $n \times n$ matrix-valued function and let λ_* be a simple eigenvalue of the* NEP (1.1). *If σ_* is the smallest singular value of $T(\lambda_*)$, and u_* and v_* are, respectively, the associated left and right unit singular vectors, then there is a neighborhood $\mathbf{N}(\lambda_*)$ of λ_* such that for all $\lambda \in \mathbf{N}(\lambda_*)$, the smallest singular value σ of $T(\lambda)$ and the corresponding left and right unit singular vectors u and v have the following properties:*

(a) $|\sigma - \sigma_*| = \mathscr{O}(|\lambda - \lambda_*|)$,

(b) $\|u - u_*\| = \mathscr{O}(|\lambda - \lambda_*|)$,

(c) $\|v - v_*\| = \mathscr{O}(|\lambda - \lambda_*|)$.

Proof. Since λ_* is a simple eigenvalue of the NEP (1.1), the rank of the matrix $T(\lambda_*)$ is $n-1$, and the smallest singular value σ_* of $T(\lambda_*)$ is also simple. According to Ref. [25], there is a neighborhood $\mathbf{N}(\lambda_*)$ of λ_* such that for all $\lambda \in \mathbf{N}(\lambda_*)$, the smallest singular value σ satisfies the relation (a). By Lemma 3.1, the smallest singular value σ and the associated left and right unit singular vectors corresponding to the eigenvalue λ_* are analytic functions so that they are Lipschitz continuous in $\mathbf{N}(\lambda_*)$, which yields relations (b) and (c). □

Lemma 3.3. *Let $T(\lambda)$ be an analytic $n \times n$ matrix-valued function and let λ_* be a simple eigenvalue of the* NEP (1.1). *Assume that σ_* is the smallest singular value of $T(\lambda_*)$, u_* and v_* are, respectively, the associated left and right unit singular vectors and λ_k, σ_k, u_k and v_k are generated by Algorithm* 2.3 *at the k-th iteration step. If $|\lambda_0 - \lambda_*| \le \varepsilon$ for an $\varepsilon \in (0,1)$, then*

(a) $\|v_k - v_*\| = \mathcal{O}(|\lambda_k - \lambda_*|)$,

(b) $\|u_k - u_*\| = \mathcal{O}(|\lambda_k - \lambda_*|)$,

(c) $\|T(\lambda_k)v_k\| = \mathcal{O}(|\lambda_k - \lambda_*|)$,

(d) $\|T^H(\lambda_k)u_k\| = \mathcal{O}(|\lambda_k - \lambda_*|)$,

for all $k = 0, 1, \cdots$.

Proof. We prove this lemma by mathematical induction. If $k = 0$, then Lemma 3.2 implies that $\sigma_* = 0$ and $\sigma_0 = u_0^H T(\lambda_0)v_0$ and the relations (a)-(d) follow. Assume now that these relations are valid for $k-1$ and examine the case for k.

Let us consider the singular value decomposition of the matrix $T(\lambda_k)$,

$$T(\lambda_k) = U(\lambda_k)\Sigma(\lambda_k)V^H(\lambda_k),$$

where $U(\lambda_k) = [u_1(\lambda_k), \cdots, u_n(\lambda_k)]$, $V(\lambda_k) = [v_1(\lambda_k), \cdots, v_n(\lambda_k)]$ are unitary matrices, and the entries of the diagonal matrix $\Sigma(\lambda_k) = \operatorname{diag}(\sigma_1(\lambda_k), \cdots, \sigma_n(\lambda_k))$ are ordered as $\sigma_1(\lambda_k) \ge \cdots \ge \sigma_n(\lambda_k) \ge 0$. It follows from Lemma 3.2 that $\sigma_n(\lambda_k) = \mathcal{O}(|\lambda_k - \lambda_*|)$. Since vectors $u_1(\lambda_k), \cdots, u_n(\lambda_k)$ constitute an orthonormal basis in $\mathbb{C}^n$, the unit vector u_{k-1} generated by Algorithm 2.3 can be represented in the form

$$u_{k-1} = \sum_{i=1}^{n} \alpha_i u_i(\lambda_k),$$

and without loss of generality, we can assume that α_n is a non-negative number. Determining x_k from the system

$$T(\lambda_k)x_k = u_{k-1} = \sum_{i=1}^{n} \alpha_i u_i(\lambda_k),$$

we obtain

$$x_k = \sum_{i=1}^{n} \alpha_i T(\lambda_k)^{-1} u_i(\lambda_k) = \sum_{i=1}^{n} \alpha_i \frac{v_i(\lambda_k)}{\sigma_i(\lambda_k)}, \tag{3.1}$$

and the norm of this vector is

$$\|x_k\| = \left(\sum_{i=1}^{n} \frac{|\alpha_i|^2}{\sigma_i^2(\lambda_k)} \right)^{1/2}.$$

Let us now show that α_n is close to 1. The induction hypothesis and Lemma 3.2 yield that

$$\|u_{k-1} - u_*\| = \mathcal{O}\big(|\lambda_{k-1} - \lambda_*|\big), \quad \big\|u_n(\lambda_k) - u_*\big\| = \mathcal{O}\big(|\lambda_k - \lambda_*|\big).$$

On the other hand,

$$\big\|u_{k-1} - u_n(\lambda_k)\big\|^2 = |\alpha_1|^2 + \cdots + |\alpha_n|^2 - 2\alpha_n + 1 = 2 - 2\alpha_n,$$

and since

$$\big\|u_{k-1} - u_n(\lambda_k)\big\| \le \|u_{k-1} - u_*\| + \big\|u_* - u_n(\lambda_k)\big\| = \mathcal{O}\big(|\lambda_{k-1} - \lambda_*|\big) + \mathcal{O}\big(|\lambda_k - \lambda_*|\big),$$

it follows that α_n is close to 1.

From (3.1), we have

$$\begin{aligned} v_n^H(\lambda_k)v_k =& \frac{v_n^H(\lambda_k)x_k}{\|x_k\|} = \frac{\alpha_n}{\sigma_n(\lambda_k)\|x_k\|} \\ =& \left(1 + \sum_{i=1}^{n-1} \frac{|\alpha_i|^2 \sigma_n^2(\lambda_k)}{\alpha_n^2 \sigma_i^2(\lambda_k)}\right)^{-1/2} = 1 + \mathcal{O}\left(\frac{\sigma_n^2(\lambda_k)}{\alpha_n^2} \sum_{i=1}^{n-1} \frac{|\alpha_i|^2}{\sigma_i^2(\lambda_k)} \right). \end{aligned} \tag{3.2}$$

Using the boundedness of the sums $\sum_{i=1}^{n-1} |\alpha_i|^2/\sigma_i^2(\lambda_k)$ and Lemma 3.2, we rewrite the relation (3.2) as

$$v_n^H(\lambda_k)v_k = 1 + \mathcal{O}\big(\sigma_n^2(\lambda_k)\big) = 1 + \mathcal{O}\big(|\lambda_k - \lambda_*|^2\big),$$

hence obtaining

$$\big\|v_n(\lambda_k) - v_k\big\|^2 = 2 - 2v_n^H(\lambda_k)v_k = \mathcal{O}\big(|\lambda_k - \lambda_*|^2\big),$$

or

$$\big\|v_n(\lambda_k) - v_k\big\| = \mathcal{O}\big(|\lambda_k - \lambda_*|\big). \tag{3.3}$$

Combining (3.3) with assertion (c) in Lemma 3.2, we obtain

$$\|v_k - v_*\| \le \big\|v_n(\lambda_k) - v_k\big\| + \big\|v_n(\lambda_k) - v_*\big\| = \mathcal{O}\big(|\lambda_k - \lambda_*|\big).$$

Let us also note that

$$\big\|T(\lambda_k)v_k\big\| = \frac{\|T(\lambda_k)x_k\|}{\|x_k\|} = \frac{\|u_{k-1}\|}{\|x_k\|} = \frac{1}{\|x_k\|} \le \frac{\sigma_n(\lambda_k)}{\alpha_n} = \mathcal{O}\big(|\lambda_k - \lambda_*|\big).$$

Since the vectors $v_1(\lambda_k),\cdots,v_n(\lambda_k)$ also constitute an orthonormal basis in $\mathbb{C}^n$, the unit vector v_k generated by Algorithm 2.3 can be represented in the form $v_k=\sum_{i=1}^n\beta_i v_i(\lambda_k)$ with a non-negative coefficient β_n. The vector y_k satisfies the equation

$$T^H(\lambda_k)y_k=v_k=\sum_{i=1}^n\beta_i v_i(\lambda_k).$$

Therefore,

$$y_k=\sum_{i=1}^n\beta_i\big(T^H(\lambda_k)\big)^{-1}v_i(\lambda_k)=\sum_{i=1}^n\beta_i\frac{u_i(\lambda_k)}{\sigma_i(\lambda_k)},$$

and its norm is

$$\|y_k\|=\left(\sum_{i=1}^n\frac{|\beta_i|^2}{\sigma_i^2(\lambda_k)}\right)^{1/2}.$$

It follows from (3.3) that

$$\left\|\sum_{i=1}^n\beta_i v_i(\lambda_k)-v_n(\lambda_k)\right\|^2=|\beta_1|^2+\cdots+|\beta_n|^2-2\beta_n+1=2-2\beta_n=\mathscr{O}\big(|\lambda_k-\lambda_*|\big),$$

or $\beta_n=1+\mathscr{O}(|\lambda_k-\lambda_*|)$. Proceeding similarly to (3.2), we obtain

$$u_n^H(\lambda_k)u_k=\frac{u_n^H(\lambda_k)y_k}{\|y_k\|}=1+\mathscr{O}\left(\frac{\sigma_n^2(\lambda_k)}{\beta_n^2}\sum_{i=1}^{n-1}\frac{|\beta_i|^2}{\sigma_i^2(\lambda_k)}\right).$$

Since the sums $\sum_{i=1}^{n-1}|\beta_i|^2/\sigma_i^2(\lambda_k)$ are bounded, we can write

$$u_n^H(\lambda_k)u_k=1+\mathscr{O}\big(\sigma_n^2(\lambda_k)\big)=1+\mathscr{O}\big(|\lambda_k-\lambda_*|^2\big),$$

which implies

$$\big\|u_n(\lambda_k)-u_k\big\|^2=2-2u_n^H(\lambda_k)u_k=\mathscr{O}\big(|\lambda_k-\lambda_*|^2\big),$$

and, subsequently,

$$\big\|u_n(\lambda_k)-u_k\big\|=\mathscr{O}\big(|\lambda_k-\lambda_*|\big).$$

Combining it with assertion (b) in Lemma 3.2, we get

$$\|u_k-u_*\|\le\big\|u_n(\lambda_k)-u_k\big\|+\big\|u_n(\lambda_k)-u_*\big\|=\mathscr{O}\big(|\lambda_k-\lambda_*|\big).$$

It follows that

$$\big\|T^H(\lambda_k)u_k\big\|=\frac{\|T^H(\lambda_k)y_k\|}{\|y_k\|}=\frac{\|v_k\|}{\|y_k\|}=\frac{1}{\|y_k\|}\le\frac{\sigma_n(\lambda_k)}{\beta_n}=\mathscr{O}\big(|\lambda_k-\lambda_*|\big),$$

which completes the proof. □

Let us now show the locally quadratic convergence of Algorithm 2.3.

Theorem 3.1. *If $u_*^H T'(\lambda_*)v_* \neq 0$, then under the assumptions of Lemma* 3.3*, the iteration sequence $\{\lambda_k\}$ generated by Algorithm* 2.3*, converges locally quadratic to the eigenvalue λ_*.*

Proof. Since the smallest singular value of $T(\lambda_*)$ is zero, one has

$$(u_k + \triangle u_k)^H T(\lambda_*)(v_k + \triangle v_k) = 0, \tag{3.4}$$

where $\triangle v_k := v_* - v_k$ and $\triangle u_k := u_* - u_k$. When $|\lambda_* - \lambda_k|$ is sufficiently small, the analytic matrix-valued function $T(\lambda)$ can be written in the form

$$T(\lambda_*) = T(\lambda_k) + (\lambda_* - \lambda_k)T'(\lambda_k) + \mathcal{O}\big(|\lambda_* - \lambda_k|^2\big). \tag{3.5}$$

From (3.4) and (3.5) one obtains

$$\begin{aligned} u_k^H\big[T(\lambda_k) + (\lambda_* - \lambda_k)T'(\lambda_k)\big]v_k = &- u_k^H\big[T(\lambda_k) + (\lambda_* - \lambda_k)T'(\lambda_k)\big]\triangle v_k \\ &- (\triangle u_k)^H\big[T(\lambda_k) + (\lambda_* - \lambda_k)T'(\lambda_k)\big]v_k \\ &+ \mathcal{O}\big(|\lambda_* - \lambda_k|^2\big). \end{aligned} \tag{3.6}$$

Moreover, according to Algorithm 2.3,

$$\lambda_{k+1} = \lambda_k - u_k^H T(\lambda_k)v_k / u_k^H T'(\lambda_k)v_k,$$

so that

$$u_k^H\big[T(\lambda_k) + (\lambda_{k+1} - \lambda_k)T'(\lambda_k)\big]v_k = 0.$$

Therefore, equation (3.6) yields

$$\begin{aligned} u_k^H\big[(\lambda_{k+1} - \lambda_*)T'(\lambda_k)\big]v_k = &u_k^H\big[T(\lambda_k) + (\lambda_* - \lambda_k)T'(\lambda_k)\big]\triangle v_k \\ &+ (\triangle u_k)^H\big[T(\lambda_k) + (\lambda_* - \lambda_k)T'(\lambda_k)\big]v_k + \mathcal{O}\big(|\lambda_* - \lambda_k|^2\big), \end{aligned}$$

and consequently,

$$\begin{aligned} \big|u_k^H T'(\lambda_k)v_k\big|\big|\lambda_{k+1} - \lambda_*\big| \leq &\big|u_k^H\big[T(\lambda_k) + (\lambda_* - \lambda_k)T'(\lambda_k)\big]\triangle v_k\big| \\ &+ \big|(\triangle u_k)^H\big[T(\lambda_k) + (\lambda_* - \lambda_k)T'(\lambda_k)\big]v_k\big| \\ &+ \mathcal{O}\big(|\lambda_* - \lambda_k|^2\big). \end{aligned} \tag{3.7}$$

Using now assertions (a)-(d) of Lemma 3.3 with $v_* - v_k = \triangle v_k$ and $u_* - u_k = \triangle u_k$, we arrive at the estimates

$$\begin{aligned} &\big|u_k^H\big[T(\lambda_k) + (\lambda_* - \lambda_k)T'(\lambda_k)\big]\triangle v_k\big| \\ \leq &\big\|T^H(\lambda_k)u_k\big\|\big\|\triangle v_k\big\| + |\lambda_* - \lambda_k|\big\|u_k^H T'(\lambda_k)\big\|\|\triangle v_k\| = \mathcal{O}\big(|\lambda_* - \lambda_k|^2\big), \\ &\big|(\triangle u_k)^H\big[T(\lambda_k) + (\lambda_* - \lambda_k)T'(\lambda_k)\big]v_k\big| \\ \leq &\big\|(\triangle u_k)^H\big\|\big\|T(\lambda_k)v_k\big\| + |\lambda_* - \lambda_k|\big\|(\triangle u_k)^H\big\|\big\|T'(\lambda_k)v_k\big\| = \mathcal{O}\big(|\lambda_* - \lambda_k|^2\big). \end{aligned} \tag{3.8}$$

However, if $|\lambda_k - \lambda_*|$ is small enough, the term $u_k^H T'(\lambda_k)v_k$ does not vanish. Therefore, relations (3.7)-(3.8) imply

$$|\lambda_{k+1} - \lambda_*| = \mathcal{O}\big(|\lambda_k - \lambda_*|^2\big).$$

which completes the proof. □

Table 1: Numerical results in Example 4.1.

	Algorithm 2.1		Algorithm 2.3	
k	λ_k	$\lvert\lambda_k-\lambda_*\rvert$	λ_k	$\lvert\lambda_k-\lambda_*\rvert$
0	1		1	
1	0.8294	2.37E+00	0.8294	2.36E+00
2	-2.6245	1.09E+00	-2.6247	1.09E+00
3	-1.8977	3.62E-01	-1.9002	3.64E-01
4	-1.5763	4.05E-02	-1.5767	4.09E-02
5	-1.5364	4.85E-04	-1.5364	4.94E-04
6	-1.5359	6.94E-08	-1.5359	7.17E-08
7	-1.5359	2.66E-15	-1.5359	1.78E-15

4. Numerical Examples

Let us compare Algorithms 2.1 and 2.3 used for the NEP (1.1). The numerical experiments have been conducted in MATLAB environment (version 2012b). The exact eigenvalue of the NEP (1.1) and the computation time (in seconds) are denoted by λ_* and CPU, respectively. The symbols σ_k and λ_k, respectively, stand for the smallest singular value and the approximated eigenvalue generated by Algorithms 2.1 and 2.3 at the k-th iteration step. The iteration procedure is terminated if $\sigma_k \le 10^{-8}$.

Example 4.1 (cf. Kressner [14]). Let us consider the nonlinear eigenvalue problem

$$T(\lambda)x = \left(\lambda A_0 - A_1 - A_2 e^{-\lambda}\right)x = 0,$$

where

$$A_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad A_1 = \begin{pmatrix} -5 & 1 \\ 2 & -6 \end{pmatrix}, \quad A_2 = \begin{pmatrix} -2 & 1 \\ 4 & -1 \end{pmatrix}.$$

The exact eigenvalue λ_* of the above problem is -1.53587607 since $\sigma_{min}(T(\lambda_*)) = 10^{-16}$.

We choose $\lambda_0 = 1$ as an initial approximation to the exact eigenvalue and find the corresponding approximations by Algorithms 2.1 and 2.3. The results are reported in Table 1. In particular, it shows that both Algorithm 2.1 and Algorithm 2.3 converge locally quadratic.

Example 4.2 (cf. Ruhe [22]). We consider the nonlinear eigenvalue problem

$$T(\lambda)x = \left((e^{\lambda} - 1)B_1 + \lambda^2 B_2 - B_0\right)x = 0,$$

where $B_0 = b_0 I$ and the matrices B_1 and B_2 are defined as follows

$$B_1 = (b_{jk}^{(1)}), \qquad b_{jk}^{(1)} = \left[n + 1 - max(j,k)\right] \times j \times k,$$
$$B_2 = (b_{jk}^{(2)}), \qquad b_{jk}^{(2)} = n\delta_{jk} + 1/(j+k).$$

For $n = 500$ and $b_0 = 500$, the exact eigenvalue λ_* of the above problem is 0.99855892 because $\sigma_{min}(T(\lambda_*)) = 2 \times 10^{-9}$.

Table 2: Numerical results in Example 4.2.

	Algorithm 2.1		Algorithm 2.3	
k	λ_k	$\lvert\lambda_k-\lambda_*\rvert$	λ_k	$\lvert\lambda_k-\lambda_*\rvert$
0	5		5	
1	2.6310	1.63E+00	2.6310	1.63E+00
2	1.5063	5.08E-01	1.5063	5.08E-01
3	1.0842	8.56E-02	1.0842	8.56E-02
4	1.0019	3.38E-03	1.0019	3.38E-03
5	0.9986	5.70E-06	0.9986	5.52E-06
6	0.9986	4.28E-10	0.9986	1.53E-11
CPU	1.260		0.323	

We choose $\lambda_0 = 5$ as an initial approximation to the exact eigenvalue and find the corresponding approximations by Algorithms 2.1 and 2.3. The results are reported in Table 2, which shows that both Algorithm 2.1 and Algorithm 2.3 converge locally quadratic and Algorithm 2.3 requires less CPU than Algorithm 2.1.

Example 4.3. We consider the nonlinear eigenvalue problem

$$T(\lambda)x = (-A_1 e^{-\lambda} - A_0 + \lambda I)x = 0,$$

where

$$A_0 = \frac{1}{h^2}\begin{pmatrix} -2 & 1 & & 0 \\ 1 & -2 & \ddots & \\ & \ddots & \ddots & 1 \\ 0 & & 1 & -2 \end{pmatrix} - 2\begin{pmatrix} \sin(h) & & & 0 \\ & \sin(2h) & & \\ & & \ddots & \\ 0 & & & \sin(nh) \end{pmatrix},$$

$$A_1 = 2\begin{pmatrix} 0 & & & \sin(h) \\ & & \sin(2h) & \\ & \iddots & & \\ \sin(nh) & & & 0 \end{pmatrix},$$

and $h = \pi/n$. For $n = 750$, the exact eigenvalue λ_* of the above problem is -0.33050859 because $\sigma_{min}(T(\lambda_*)) = 3 \times 10^{-11}$.

We choose $\lambda_0 = 0$ as an initial approximation to the exact eigenvalue and find the corresponding approximations by Algorithms 2.1 and 2.3. The results are reported in Table 3, which shows that both Algorithm 2.1 and Algorithm 2.3 converge locally quadratic and Algorithm 2.3 requires less CPU than Algorithm 2.1.

5. Conclusion

We propose a modification to the Newton method for nonlinear eigenvalue problems and establish its locally quadratic convergence. Numerical examples show the efficiency and lower computational cost of this approach.

Table 3: Numerical results in Example 4.3.

	Algorithm 2.1		Algorithm 2.3	
k	λ_k	$\|\lambda_k - \lambda_*\|$	λ_k	$\|\lambda_k - \lambda_*\|$
0	0		0	
1	-0.3718	4.13E-02	-0.3718	4.13E-02
2	-0.3311	6.12E-04	-0.3311	6.14E-04
3	-0.3305	1.35E-07	-0.3305	1.36E-07
4	-0.3305	2.32E-13	-0.3305	1.11E-16
CPU	2.6411		0.8274	

Acknowledgments

The authors thank Professor Roger Hosking, Professor Zhong-Zhi Bai and anonymous referees for useful comments and suggestions which helped to improve the presentation of this paper. The authors are also grateful to Professor Victor D. Didenko for his careful editing. The research was supported by the National Natural Science Foundation of China (Grants Nos.11571171 and 11701409), the Natural Science Foundation of Jiangsu Province (Grant No.BK20170591), the Natural Science Foundation of Jiangsu Higher Education Institutions of China (Grant No.17KJB110018).

References

[1] S. Adhikari and B. Pascual, *Eigenvalues of linear viscoelastic systems*, J. Sound Vib. **325**, 1000–1011 (2009).

[2] A. Bunse-Gerstner, R. Byers, V. Mehrmann and N. K. Nichols, *Numerical computation of an analytic singular value decomposition of a matrix valued function*, Numer. Math. **60**, 1–39 (1991).

[3] C. Conca, A. Osses and J. Planchard, *Asymptotic analysis relating spectral models in fluid-solid vibrations*, SIAM J. Numer. Anal. **35**, 1020–1048 (1998).

[4] H. Dai and Z. Z. Bai, *On smooth LU decompositions with applications to solutions of nonlinear eigenvalue problems*, J. Comput. Math. **28**, 745–766 (2010).

[5] E. M. Daya and M. Potier-Ferry, *A numerical method for nonlinear eigenvalue problems application to vibrations of viscoelastic structures*, Comput. Struct. **79**, 533–541 (2001).

[6] I. Gohberg, P. Lancaster and L. Rodman, *Matrix Polynomials*, Academic Press (1982).

[7] J. S. Guo, W. W. Lin and C. S. Wang, *Nonequivalence deflation for the solution of matrix talent value problems*, Linear Algebra Appl. **231**, 15–45 (1995).

[8] T. M. Hwang, W. W. Lin, W. C. Wang and W. Wang, *Numerical simulation of three dimensional pyramid quantum dot*, J. Comput. Phys. **196**, 208–232 (2004).

[9] N. K. Jain and K. Singhal, *On Kublanovskaya's approach to the solution of the generalized latent value problem for functional λ-matrices*, SIAM J. Numer. Anal. **20**, 1062–1070 (1983).

[10] E. Jarlebring, *The spectrum of delay-differential equations: numerical methods, stability and perturbation*, Ph.D. Thesis, TU Braunschweig 2008.

[11] E. Jarlebring, K. Meerbergen and W. Michiels, *A Krylov method for the delay eigenvalue problem*, SIAM J. Sci. Comput. **32**, 3278–3300 (2010).

[12] T. Kato, *Perturbation Theory for Linear Operators*, 2nd edn., Springer-Verlag, Berlin, 1976.

[13] V. B. Khazanov and V. N. Kublanovskaya, *Spectral problems for matrix pencils: Methods and algorithms* II, Sov. J. Numer. Anal. Math. Modelling, **3**, 467–485 (1988).

[14] D. Kressner, *A block Newton method for nonlinear eigenvalue problems*, Numer. Math. **114**, 355–372 (2009).

[15] V. N. Kublanovskaya, *On an approach to the solution of the generalized latent value problem for λ-matrices*, SIAM J. Numer. Anal. **7**, 532–537 (1970).

[16] P. Lancaster, *Lambda-Matrices and Vibrating Systems*, Pergamon Press (1966).

[17] R. C. Li, *QR decomposition and nonlinear eigenvalue problems*, Math. Numer. Sinica, **11**, 374–385 (1989).

[18] R. C. Li, *Compute multiply nonlinear eigenvalues*, J. Comput. Math. **10**, 1–20 (1992).

[19] V. Mehrmann and H. Voss, *Nonlinear eigenvalue problems: A challenge for modern eigenvalue methods*, GAMM Mitteilungen, **27**, 121–152 (2004).

[20] S. I. Niculescu, *Delay Effects on Stability: A Robust Control Approach*, Springer (2001).

[21] G. Peters and J. H. Wilkinson, *Inverse iteration, ill-conditioned equations and Newton's method*, SIAM Rev. **21**, 339–360 (1979).

[22] A. Ruhe, *Algorithms for the nonlinear eigenvalue problems*, SIAM J. Numer. Anal. **10**, 674–689 (1973).

[23] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd edn., SIAM (2003).

[24] A. Spence and C. Poulton, *Photonic band structure calculations using nonlinear eigenvalue techniques*, J. Comput. Phys. **204**, 65–81 (2005).

[25] J. G. Sun, *Matrix Perturbation Theory*, 2nd edn., Science Press (2001).

[26] W. Sun and K. M. Liu, *Iterative algorithms for nonlinear ordinary differential eigenvalue problems*, Appl. Numer. Math. **38**, 361–376 (2001).

[27] F. Tisseur and K. Meerbergen, *The quadratic eigenvalue problem*, SIAM Rev. **43**, 235–286 (2001).

[28] S. Van Huffel, J. Vandewalle and A. Haegemans, *An efficient and reliable algorithm for computing the singular subspace of a matrix, associated with its smallest singular values*, J. Comput. Appl. Math. **19**, 313-333 (1987).

[29] J. M. Varah, *The calculation of the eigenvectors of a general complex matrix by inverse iteration*, Math. Comput. **22**, 785–791 (1968).

[30] H. Voss, *A rational spectral problem in fluid-solid vibration*, Electron. Trans. Numer. Anal. **16**, 94–106 (2003).

[31] H. Voss, *Numerical calculation of the electronic structure for three-dimensional quantum dots*, Comput. Phys. Commun. **174**, 441–446 (2006).

[32] R. Wobst, *The generalized eigenvalue problem and acoustic surface wave computations*, Computing, **39**, 57–69 (1987).

[33] K. Wright, *Differential equations for the analytic singular value decomposition of a matrix*, Numer. Math. **63**, 283–295 (1992).

[34] S. F. Xu, *A smallest singular value method for solving inverse eigenvalue problems*, J. Comput. Math. **14**, 23–31 (1996).