NUMERICAL METHODS FOR THE TRIDIAGONAL HYPERBOLIC QUADRATIC EIGENVALUE PROBLEM[*]

BOR PLESTENJAK[†]

Abstract. We consider numerical methods for the computation of the eigenvalues of the tridiagonal hyperbolic quadratic eigenvalue problem. The eigenvalues are computed as zeros of the characteristic polynomial using the bisection, Laguerre's method, and the Ehrlich–Aberth method. Initial approximations are provided by a divide-and-conquer approach using rank two modifications. The above methods need a stable and efficient evaluation of the quadratic eigenvalue problem's characteristic polynomial and its derivatives. We discuss how to obtain these values using three-term recurrences, the QR factorization, and the LU factorization.

Key words. Quadratic eigenvalue problem, inertia, Laguerre's method, Ehrlich–Aberth method, bisection, LU factorization, QR factorization, divide-and-conquer

AMS subject classifications. 65F15, 15A18, 15A69.

1. Introduction. We consider a Hermitian quadratic eigenvalue problem (QEP)

$$(1.1) \qquad Q(\lambda)x = (\lambda^2 M + \lambda C + K)x = 0,$$

where M, C, and K are $n \times n$ Hermitian matrices. If (1.1) is satisfied for a nonzero $x \in \mathbb{C}^n$ and $\lambda \in \mathbb{C}$, then λ is an eigenvalue and x is the corresponding (right) eigenvector. The characteristic polynomial $f(\lambda) = \det(Q(\lambda))$ is of degree less than or equal to $2n$. A QEP is regular when f is not identically zero. A regular QEP has $2n$ finite or infinite eigenvalues. The finite eigenvalues are the zeros of f while the infinite eigenvalues correspond to the zero eigenvalues of the reversed QEP $\lambda^2 Q(1/\lambda) = \lambda^2 K + \lambda C + M$. If M is nonsingular then we have $2n$ finite eigenvalues with up to $2n$ eigenvectors, which are not necessarily linearly independent. QEPs appear in various applications, for a recent survey of the QEP see [19].

We say that a QEP is *hyperbolic* [11] if M is positive definite and

$$(x^*Cx)^2 > 4(x^*Mx)(x^*Kx)$$

for all $x \neq 0$. For a hyperbolic QEP the eigenvalues are all real. In this paper we focus on the real tridiagonal hyperbolic QEP, where matrices M, C, and K are symmetric and tridiagonal. An example of a tridiagonal quadratic eigenvalue problem is a damped mass-spring system (see, e.g., [19]). Our goal is to compute all or some of the eigenvalues. For the computation of the eigenvalues we apply polynomial solvers to the characteristic polynomial. The eigenvectors can be later obtained by the inverse iteration, for a stable algorithm see [6]. We show that some of the presented methods can be applied to more general problems, e.g., to the banded polynomial eigenvalue problems, as well.

The paper is organized as follows. In Section 2 we recall some results on the hyperbolic QEPs. The inertia of a hyperbolic QEP is discussed in Section 3. In Sections 4, 5, and 6 three different approaches, based respectively on the three-term recurrences, QR factorization, and LU factorization, for the computation of the derivatives of the characteristic polynomial are presented. The divide-and-conquer approach for the initial approximations is presented in Section 7. In Sections 8 and 9 Laguerre's method and the Ehrlich–Aberth method are applied

to the computation of the zeros of the characteristic polynomial, respectively. Some numerical examples are given in Section 10, followed by conclusions.

2. Auxiliary results. The following properties of the hyperbolic QEPs are gathered from [7, 11, 16]. A hyperbolic QEP has $2n$ real eigenvalues and eigenvectors. All eigenvalues are semisimple and there is a gap between the largest n (*primary*) and the smallest n (*secondary*) eigenvalues. There are n linearly independent eigenvectors associated with the primary and the secondary eigenvalues, respectively.

For each $x \neq 0$ the equation

$$\mu^2 x^T M x + \mu x^T C x + x^T K x = 0$$

has two distinct real solutions $\mu_1(x) < \mu_2(x)$. If x is an eigenvector, then at least one of $\mu_1(x)$ and $\mu_2(x)$ is the corresponding eigenvalue. Values $\mu_1(x)$ and $\mu_2(x)$ are generalizations of the Rayleigh quotient and similar to the symmetric matrices there exists minimax theorem for the hyperbolic QEP as well.

THEOREM 2.1 (Duffin [7]). *If* $\lambda_{2n} \leq \cdots \leq \lambda_1$ *are the eigenvalues of a hyperbolic QEP then*

$$\lambda_{n+i} = \max_{\substack{S \subset \mathbb{R}^n \\ \dim(S) = i}} \min_{0 \neq x \in S} \mu_1(x) \qquad \text{and} \qquad \lambda_i = \max_{\substack{S \subset \mathbb{R}^n \\ \dim(S) = i}} \min_{0 \neq x \in S} \mu_2(x)$$

for $i = 1, \ldots, n$.

THEOREM 2.2 (Markus [16]). *A Hermitian QEP where* M *is positive definite is hyperbolic if and only if there exists* $\gamma \in \mathbb{R}$ *such that the matrix* $Q(\gamma)$ *is negative definite.*

REMARK 2.3. *The scalar* γ *in Theorem 2.2, such that* $Q(\gamma)$ *is negative definite, lies in the gap between the primary and the secondary eigenvalues, i.e.,* $\lambda_{n+1} < \gamma < \lambda_n$.

3. Inertia of a hyperbolic QEP. The inertia of a symmetric matrix A is a triplet of nonnegative integers (ν, ζ, π), where $\nu, \zeta,$ and π are respectively the number of negative, zero and positive eigenvalues of A. The following theorem shows that the inertia of a symmetric matrix $Q(\sigma)$ is related to the number of eigenvalues of the QEP Q that are larger or smaller than σ, respectively.

THEOREM 3.1. *Let* $M, C,$ *and* K *be symmetric* $n \times n$ *matrices such that* $Q(\lambda) = \lambda^2 M + \lambda C + K$ *is a hyperbolic QEP and let* $\lambda_{2n} \leq \cdots \leq \lambda_{n+1} < \lambda_n \leq \cdots \leq \lambda_1$ *be the eigenvalues of the QEP* Q. *If* (ν, ζ, π) *is the inertia of the matrix* $Q(\sigma)$ *then* ζ *is the algebraic multiplicity of* σ *as an eigenvalue of the QEP* Q *and*
  a) *if* $\sigma > \lambda_n$ *then* ν *is the number of eigenvalues of* Q *larger than* σ *and* $\pi + n$ *is the number of eigenvalues of* Q *smaller than* σ,
  b) *if* $\sigma < \lambda_{n+1}$ *then* ν *is the number of eigenvalues of* Q *smaller than* σ *and* $\pi + n$ *is the number of eigenvalues of* Q *larger than* σ.

Proof. For each $\lambda \in \mathbb{R}$, $Q(\lambda)$ is a symmetric $n \times n$ matrix with n real ordered eigenvalues

$$(3.1) \qquad\qquad\qquad \mu_n(\lambda) \leq \cdots \leq \mu_1(\lambda),$$

where $\mu_1, \ldots, \mu_n$ are continuous functions of λ. It is easy to see that σ is an eigenvalue of the QEP Q of algebraic multiplicity k exactly when there exists $1 \leq i \leq n$ such that

$$\mu_i(\sigma) = \mu_{i+1}(\sigma) = \cdots = \mu_{i+k-1}(\sigma) = 0.$$

Since M is a symmetric positive definite matrix,

$$\lim_{\lambda \to \pm\infty} \mu_i(\lambda) = \infty$$

for all i. By Theorem 2.2 there exists $\sigma_0 \in \mathbb{R}$ such that $\mu_i(\sigma_0) < 0$ for all i. Because each μ_i is a continuous function it has at least two zeros, one on the right and one on the left side of σ_0. As each zero of μ_i is also an eigenvalue of the QEP Q which has $2n$ eigenvalues, it follows that each μ_i has exactly two zeros.

As $\mu_1, \ldots, \mu_n$ are continuous and ordered as in (3.1), it is not hard to deduce that if $\sigma > \sigma_0$ and σ is not an eigenvalue of Q then the number of negative eigenvalues of $Q(\sigma)$ equals the number of eigenvalues of Q that are larger than σ. This proves a) and similarly we can prove b). $\square$

REMARK 3.2. *Theorem 3.1 is a generalization of a similar theorem in [18], where M is a positive definite matrix and K is a negative definite matrix. In this case $Q(0)$ is negative definite and a proof similar to the above can be done without applying Theorem 2.2.*

Based on the inertia we could apply the bisection to obtain the kth eigenvalue. The algorithm is similar to the algorithm for the symmetric eigenvalue problem. To derive more efficient methods, we use some faster methods that were successfully applied to tridiagonal eigenvalue problems: Laguerre's method [14, 15] and the Ehrlich–Aberth method [4].

The above methods need stable and efficient computation of $\nu(Q(\lambda))$, $f(\lambda)$, $f'(\lambda)/f(\lambda)$ and $f''(\lambda)/f(\lambda)$, where $f(\lambda) = \det(Q(\lambda))$. We discuss how to obtain these values using the three-term recurrences, the QR factorization, and the LU factorization in the next three sections.

4. Three-term recurrences. Let $Q(\lambda) = (\lambda^2 M + \lambda C + K)$, where M, C, and K are $n \times n$ tridiagonal matrices. We can write

$$
Q(\lambda) = \begin{bmatrix}
a_1(\lambda) & b_1(\lambda) & & & 0 \\
b_1(\lambda) & a_2(\lambda) & b_2(\lambda) & & \\
& \ddots & \ddots & \ddots & \\
& & b_{n-2}(\lambda) & a_{n-1}(\lambda) & b_{n-1}(\lambda) \\
0 & & & b_{n-1}(\lambda) & a_n(\lambda)
\end{bmatrix},
$$

where $a_i(\lambda) = \lambda^2 M_{ii} + \lambda C_{ii} + K_{ii}$ and $b_i(\lambda) = \lambda^2 M_{i+1,i} + \lambda C_{i+1,i} + K_{i+1,i}$ are quadratic polynomials. The determinant of a tridiagonal matrix can be computed using a three-term recurrence, see, e.g., [9]. If $f_k(\lambda) = \det(Q_k(\lambda))$, where $Q_k(\lambda)$ is the leading $k \times k$ submatrix of $Q(\lambda)$, then

$$
\begin{array}{rclcrcl}
f_0 & = & 1, & & f_1 & = & a_1, \\
f_0' & = & 0, & & f_1' & = & a_1', \\
f_0'' & = & 0, & & f_1'' & = & a_1'',
\end{array}
$$

and

$$
\begin{aligned}
f_{r+1} &= a_{r+1} f_r - b_r^2 f_{r-1}, \\
f_{r+1}' &= a_{r+1}' f_r + a_{r+1} f_r' - 2 b_r b_r' f_{r-1} - b_r^2 f_{r-1}', \\
f_{r+1}'' &= a_{r+1}'' f_r + 2 a_{r+1}' f_r' + a_{r+1} f_r'' - 2 b_r'^2 f_{r-1} - 2 b_r b_r'' f_{r-1} - 4 b_r b_r' f_{r-1}' - b_r^2 f_{r-1}''
\end{aligned}
$$

for $r = 1, \ldots, n - 1$. For the sake of brevity the argument λ is omitted in the above equations.

As the above recurrences may suffer from overflow and underflow problems [13], we define

$$
d_i = \frac{f_i}{f_{i-1}}, \quad g_i = \frac{f_i'}{f_i}, \quad h_i = \frac{f_i''}{f_i}.
$$

Then $f_n = d_1 \cdots d_n$,

$$
\begin{aligned}
d_1 &= a_1, \\
g_0 &= 0, & g_1 &= \frac{a_1'}{a_1}, \\
h_0 &= 0, & h_1 &= \frac{a_1''}{a_1},
\end{aligned}
$$

and

$$
d_{r+1} = a_{r+1} - \frac{b_r^2}{d_r},
$$
$$
g_{r+1} = \frac{1}{d_{r+1}} \left(a_{r+1}' + a_{r+1} g_r - \frac{1}{d_r}(2 b_r b_r' + b_r^2 g_{r-1}) \right),
$$
$$
h_{r+1} = \frac{1}{d_{r+1}} \left(a_{r+1}'' + 2 a_{r+1}' g_r + a_{r+1} h_r - \frac{1}{d_r}(2 b_r'^2 + 2 b_r b_r'' + 4 b_r b_r' g_{r-1} + b_r^2 h_{r-1}) \right)
$$

for $r = 1, \ldots, n-1$.

REMARK 4.1. *One can see that $d_1, \ldots, d_n$ are the diagonal elements from the LDL^T factorization of the matrix $Q(\lambda)$.*

REMARK 4.2. *The algorithm may break down if $d_r = 0$ for some $r = 1, \ldots, n-1$. In such case we introduce small perturbations and set*

$$
d_r = \frac{\varepsilon}{d_{r-1}} \left(|\lambda|^2 |M_{r-1,r-1}| + |\lambda||C_{r-1,r-1}| + |K_{r-1,r-1}| + \varepsilon \right),
$$

where ε is the machine precision. This corresponds to a small relative perturbation of the matrices M, C, and K. A similar approach is used in [14].

5. A QR factorization approach. If $f(\lambda) \neq 0$ then it follows from Jacobi's formula for the derivative of the determinant that

(5.1) $$ f'(\lambda)/f(\lambda) = \operatorname{tr}(Q(\lambda)^{-1} Q'(\lambda)). $$

If we denote $A = Q(\lambda)$ and $B = Q'(\lambda)$, then we need to compute $\operatorname{tr}(A^{-1}B)$, where in our case A and B are tridiagonal matrices. In [4] one can find a stable $\mathcal{O}(n)$ computation of $\operatorname{tr}(A^{-1})$ via QR factorization. In this section we generalize this algorithm to compute $\operatorname{tr}(A^{-1}B)$. We start with a sketch of the algorithm for $\operatorname{tr}(A^{-1})$, for details and the theory, see [4].

Let A be a tridiagonal matrix and let $A = UR$, where

$$
R = \begin{bmatrix}
r_1 & s_1 & t_1 & & & \\
& \ddots & \ddots & \ddots & & \\
& & r_{n-2} & s_{n-2} & t_{n-2} \\
& & & r_{n-1} & s_{n-1} \\
& & & & r_n
\end{bmatrix}
$$

is an upper triangular tridiagonal matrix and U is the product of $n-1$ Givens rotations, $U^* = G_{n-1} \cdots G_2 G_1$, where

$$
G_i([i, i+1], [i, i+1]) = \begin{bmatrix} \psi_i & \theta_i \\ -\overline{\theta}_i & \overline{\psi}_i \end{bmatrix} \quad \text{and} \quad |\psi_i|^2 + |\theta_i|^2 = 1.
$$

Then

$$U^* = \begin{bmatrix} v_1 u_1 & \psi_1 & & & 0 \\ v_2 u_1 & v_2 u_2 & \psi_2 & & \\ \vdots & \vdots & \ddots & \ddots & \\ \vdots & & & v_{n-1} u_{n-1} & \psi_{n-1} \\ v_n u_1 & v_n u_2 & \cdots & v_n u_{n-1} & v_n u_n \end{bmatrix},$$

where

$$D = \mathrm{diag}(1, -\overline{\psi_1}, \overline{\psi_1}\,\overline{\psi_2}, \ldots, (-1)^{n-1}\overline{\psi_1}\,\overline{\psi_2}\cdots\overline{\psi_{n-1}}),$$
$$u = D^{-1}[1, \overline{\theta_1}, \cdots, \overline{\theta_{n-1}}]^T,$$
$$v = D[\theta_1, \cdots, \theta_{n-1}, 1]^T.$$

If we solve $Rw = v$, then

(5.2)
$$\mathrm{tr}(A^{-1}) = \sum_{i=1}^{n} u_i w_i.$$

Kressner [10] generalized the above approach into an $\mathcal{O}(n)$ algorithm for the computation of $\mathrm{tr}(A^{-1}B)$, where both matrices A and B are tridiagonal. Suppose that

$$B = \begin{bmatrix} x_1 & z_1 & & & 0 \\ y_1 & x_2 & z_2 & & \\ & \ddots & \ddots & \ddots & \\ & & y_{n-2} & x_{n-1} & z_{n-1} \\ 0 & & & y_{n-1} & x_n \end{bmatrix}.$$

To compute $\mathrm{tr}(A^{-1}B)$ we need the diagonal elements of $A^{-1}B$. From

$$(A^{-1}B)_{ii} = e_i^T R^{-1} U^* B e_i$$
$$= z_{i-1} e_i^T R^{-1} U^* e_{i-1} + x_i e_i^T R^{-1} U^* e_i + y_i e_i^T R^{-1} U^* e_{i+1},$$

and

$$e_i^T R^{-1} U^* e_{i-1} = u_{i-1} w_i,$$

$$e_i^T R^{-1} U^* e_i \quad = u_i w_i,$$

$$e_i^T R^{-1} U^* e_{i+1} = u_{i+1} w_i + \frac{1}{r_i}(\psi_i - v_i u_{i+1})$$

it follows that

(5.3)
$$\mathrm{tr}(A^{-1}B) = \sum_{i=2}^{n} z_{i-1} u_{i-1} w_i + \sum_{i=1}^{n} x_i u_i w_i + \sum_{i=1}^{n-1} y_i \left(u_{i+1} w_i + \frac{1}{r_i}(\psi_i - v_i u_{i+1}) \right).$$

As reported in [4], formula (5.2) is not stable. To make it stable, we have to avoid the explicit multiplication by the matrix D or D^{-1}. If we define $\hat{R} = D^{-1}RD$, $\hat{v} = D^{-1}v$, $\hat{u} = Du$, and solve $\hat{R}\hat{w} = \hat{v}$ for $\hat{w}$, then

(5.4)
$$\mathrm{tr}(A^{-1}) = \sum_{i=1}^{n} \hat{u}_i \hat{w}_i.$$

Notice that

$$\hat{R} = \begin{bmatrix} \hat{r}_1 & \hat{s}_1 & \hat{t}_1 & & \\ & \ddots & \ddots & \ddots & \\ & & \hat{r}_{n-2} & \hat{s}_{n-2} & \hat{t}_{n-2} \\ & & & \hat{r}_{n-1} & \hat{s}_{n-1} \\ & & & & \hat{r}_n \end{bmatrix},$$

where $\hat{r}_i = r_i$, $\hat{s}_i = -\overline{\psi_i}s_i$, and $\hat{t}_i = -\overline{\psi_i\psi_{i+1}}t_i$.

Using the same notation it follows from

$$\begin{aligned} u_i w_i &= \hat{u}_i \hat{w}_i, \\ u_{i-1} w_{i-1} &= -\overline{\psi_{i-1}}\hat{u}_{i-1}\hat{w}_i, \\ u_{i+1} w_{i-1} &= -\hat{u}_{i+1}\hat{w}_i(\overline{\psi_i})^{-1}, \\ v_i u_{i+1} &= -\hat{v}_i \hat{u}_{i+1}(\overline{\psi_i})^{-1} \end{aligned}$$

that we may rewrite formula (5.3) in a stable form

$$\operatorname{tr}(A^{-1}B) = \sum_{i=2}^{n} x_i \hat{u}_i \hat{w}_i - \sum_{i=1}^{n} z_{i-1}\overline{\psi_{i-1}}\hat{u}_{i-1}\hat{w}_i - \sum_{i=1}^{n-1} \frac{y_i}{\overline{\psi_i}}\left(\hat{u}_{i+1}\hat{w}_i + \frac{1}{r_i}(|\psi_i|^2 + \hat{v}_i\hat{u}_{i+1})\right).$$

6. An LU factorization approach. In [5] one can find an algorithm for the computation of the derivative of the determinant using the LU factorization. Suppose that $\det(Q(\lambda)) \neq 0$ and that $PQ(\lambda) = LU$ is the result of Gaussian elimination with partial pivoting for $Q(\lambda)$, where L is a lower triangular matrix with ones on the diagonal and U is an upper triangular matrix. Then

$$f(\lambda) = \det(Q(\lambda)) = \det(P) \cdot u_{11}u_{22}\cdots u_{nn}.$$

If we fix the permutation matrix P then for each μ in a small neighborhood of λ there exist analytic matrices $L(\mu)$ and $U(\mu)$ such that

(6.1) $L(\mu)U(\mu) = PQ(\mu)$

is the LU factorization of $PQ(\mu)$. By differentiating (6.1) at $\mu = \lambda$ we get

$$PQ' = L'U + LU' = MU + LV,$$

where $M = L'$ is a lower triangular matrix with zeros on the diagonal and $V = U'$ is an upper triangular matrix. Matrices M and V of the proper form and such that $PQ' = MU + LV$ can be computed from $Q', P, L,$ and U (see Algorithm 6.1). It follows that

$$f'(\lambda) = \det(P) \sum_{i=1}^{n} v_{ii} \prod_{\substack{j=1 \\ j \neq i}}^{n} u_{jj}$$

and

$$\frac{f'(\lambda)}{f(\lambda)} = \sum_{i=1}^{n} \frac{v_{ii}}{u_{ii}}.$$

ALGORITHM 6.1 (Bohte [5]). *The algorithm solves the equation $B = MU + LV$ for M and V, where L is a lower triangular matrix with ones on the main diagonal, U is a nonsingular upper triangular matrix, B is a square $n \times n$ matrix, M is a lower triangular matrix with zeros on the main diagonal, and V is an upper triangular matrix.*

$$
\begin{aligned}
&\text{for } r = 1 \text{ to } n \\
&\quad \text{for } k = r \text{ to } n \\
&\qquad v_{rk} = b_{rk} - \sum_{j=1}^{r-1}(m_{rj}u_{jk} + l_{rj}v_{jk}) \\
&\quad \text{for } i = r + 1 \text{ to } n \\
&\qquad m_{ir} = \frac{1}{u_{rr}}\left(b_{ir} - \sum_{j=1}^{r-1}(m_{ij}u_{jr} + l_{ij}v_{jr}) - l_{ir}v_{rr}\right)
\end{aligned}
$$

For the second derivative we have

(6.2) $$PQ'' = L''U + 2L'U' + LU'' = NU + 2MV + LW,$$

where $N = L''$ is a lower triangular matrix with zeros on the diagonal and $W = U''$ is an upper triangular matrix. It follows that

$$\frac{f''(\lambda)}{f(\lambda)} = \sum_{i=1}^{n} \frac{w_{ii}}{u_{ii}} + \left(\sum_{i=1}^{n} \frac{v_{ii}}{u_{ii}}\right)^2 - \sum_{i=1}^{n} \frac{v_{ii}^2}{u_{ii}^2}.$$

From the relation (6.2) we get $PQ'' - 2MV = NU + LW$, which means that we can apply Algorithm 6.1 for the computation of N and W as well.

An implementation of Algorithm 6.1 for banded matrices computes f'/f and f''/f in a linear time. The algorithm is more expensive than the three-term recurrences in Section 4, but its advantage is that it can be applied to non-tridiagonal matrices as well. Let us also mention that in [5] one can find a slightly modified algorithm that is able to compute $f'(\lambda)$ even if $f(\lambda) = 0$.

7. Divide-and-conquer. We choose $m \approx n/2$ and write

$$Q(\lambda) = Q_0(\lambda) + b_m(\lambda)(e_{m-1}e_{m+1}^T + e_{m+1}e_{m-1}^T),$$

where

$$Q_0(\lambda) = \begin{bmatrix} Q_1(\lambda) & 0 \\ 0 & Q_2(\lambda) \end{bmatrix}.$$

$Q_0(\lambda)$ is a rank two modification of $Q(\lambda)$. If we apply Theorem 2.2 then it is not hard to see that Q_1 and Q_2 are hyperbolic QEPs. The eigenvalues $\widetilde{\lambda}_{2n} \leq \cdots \leq \widetilde{\lambda}_1$ of Q_0, a union of the eigenvalues of Q_1 and Q_2, are approximations to the eigenvalues $\lambda_{2n} \leq \cdots \leq \lambda_1$ of Q.

We can show that the eigenvalues of Q_0 and Q interlace. To show this useful property we introduce a convex combination of Q_0 and Q. Let Q_t be a QEP defined by

$$Q_t(\lambda) = (1 - t)Q_0(\lambda) + tQ(\lambda).$$

LEMMA 7.1. *The QEP Q_t is hyperbolic for $t \in [0, 1]$.*

Proof. From Theorem 2.2 it follows that there exists γ such that $Q(\gamma)$ is negative definite. Being principal submatrices of $Q(\gamma)$, matrices $Q_1(\gamma)$ and $Q_2(\gamma)$ are negative definite as well. Since it is a block diagonal matrix with negative definite blocks $Q_1(\gamma)$ and $Q_2(\gamma)$, matrix $Q_0(\gamma)$

is negative definite, too. For $t \in [0,1]$ it is now easy to see that $Q_t(\gamma) = (1-t)Q_0 + tQ(\gamma)$ is negative definite and Theorem 2.2 yields that Q_t is a hyperbolic QEP. $\square$

The following theory is a generalization of Theorem 5.2 in [15].

LEMMA 7.2. *Let $\lambda_{2n}(t) \leq \cdots \leq \lambda_1(t)$ be the ordered eigenvalues of the QEP Q_t for $t \in [0,1]$. Each eigencurve $\lambda_i(t)$ is then either constant or strictly monotone for $t \in [0,1]$ and $i = 1, \ldots, 2n$. If we define $\widetilde{\lambda}_0 = \infty$ and $\widetilde{\lambda}_{2n+1} = \infty$ then each eigencurve $\lambda_i(t)$ lies on the interval $[\widetilde{\lambda}_{i+1}, \widetilde{\lambda}_{i-1}]$.*

Proof. From the construction of Q_t (see, for example, the three-term recurrences in Section 4) it follows that the determinant of $Q_t(\lambda)$ can be expressed as

$$\det Q_t(\lambda) = p_1(\lambda) + t^2 p_2(\lambda),$$

where p_1 and p_2 are polynomials of degree $2n$.

If for a chosen λ_0 we have $p_2(\lambda_0) \neq 0$, then the equation $p(t, \lambda_0) = 0$ has at most one solution on $(0,1]$. If $p_2(\lambda_0) = 0$ and $p_1(\lambda_0) \neq 0$, then none of the eigencurves passes the line $\lambda = \lambda_0$. If $p_2(\lambda_0) = 0$ and $p_1(\lambda_0) = 0$, then λ_0 is an eigenvalue of Q_t for $t \in [0,1]$ and at least one eigencurve $\lambda_i(t)$ is constant and equal to λ_0.

It follows from the above that the eigencurves $\lambda_i(t)$ for $i = 1, \ldots, 2n$ are either constant or strictly monotone for $t \in [0,1]$ (see Figure 7.1). For each $\widetilde{\lambda}_i$ either the only solution of $p(t, \widetilde{\lambda}_i) = 0$ is at $t = 0$ or the eigencurve $\lambda_i(t)$ is constant and equal to $\widetilde{\lambda}_i$. Therefore, $\lambda_i(t)$ is bounded below and above by $\widetilde{\lambda}_{i+1}$ and $\widetilde{\lambda}_{i-1}$, respectively. $\square$
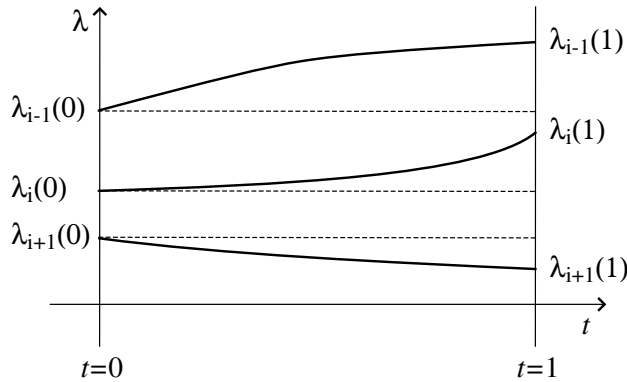

FIG. 7.1. *Eigenvalues of Q_0 and Q interlace.*

THEOREM 7.3. *Let $\widetilde{\lambda}_{2n} \leq \cdots \leq \widetilde{\lambda}_1$ be the eigenvalues of $Q_0(\lambda)$ and $\lambda_{2n} \leq \cdots \leq \lambda_1$ the eigenvalues of $Q(\lambda)$. Then:*
  a) $\widetilde{\lambda}_1 \leq \lambda_1$ and $\lambda_{2n} \leq \widetilde{\lambda}_{2n}$,
  b) $\widetilde{\lambda}_{i+1} \leq \lambda_i \leq \widetilde{\lambda}_{i-1}$ for $i = 2, \ldots, n-1$ and $i = n+2, \ldots, 2n-1$,
  c) $\widetilde{\lambda}_{n+1} \leq \lambda_{n+1} < \lambda_n \leq \widetilde{\lambda}_n$.

Proof. As matrices $Q_1(\lambda)$ and $Q_2(\lambda)$ are submatrices of $Q(\lambda)$, it follows from Theorem 2.1 that the primary eigenvalues of Q_1 and Q_2 lie in the interval $[\lambda_n, \lambda_1]$. Because of that the primary eigenvalues of Q_0 lie in the interval $[\lambda_n, \lambda_1]$. Similarly we can show that the secondary eigenvalues of Q_0 lie in the interval $[\lambda_{2n}, \lambda_{n+1}]$. Thus we prove points a) and c).

Point b) follows from Lemma 7.2. We know that λ_i and $\widetilde{\lambda}_i$ are connected by a monotone eigencurve $\lambda_i(t)$, which is bounded below and above by $\widetilde{\lambda}_{i+1}$ and $\widetilde{\lambda}_{i-1}$, respectively. $\square$

REMARK 7.4. *Unlike the divide-and-conquer method for the symmetric tridiagonal matrices, here $\widetilde{\lambda}_i = \widetilde{\lambda}_{i+1}$ does not imply that one of the eigenvalues of Q is $\widetilde{\lambda}_i$. Only if $\widetilde{\lambda}_{i-1} = \widetilde{\lambda}_i = \widetilde{\lambda}_{i+1}$ then one can deduce that $\widetilde{\lambda}_i$ is an eigenvalue of Q.*

In the conquer phase we use a numerical method that computes the eigenvalues $\lambda_1, \ldots, \lambda_{2n}$ of the QEP Q from the initial approximations $\widetilde{\lambda}_1, \ldots, \widetilde{\lambda}_{2n}$. Two numerical methods that may be applied for this task are presented in the next two sections. We are not claiming that these are the optimal methods. Other polynomial solvers applied to the classical or to the generalized eigenvalue problem with tridiagonal matrices (see, e.g., [12, 17]) could be applied to the QEP as well.

8. Laguerre's method. To the characteristic polynomial $f(\lambda) = \det(Q(\lambda))$ we can apply Laguerre's method, a well-known globally convergent method for finding polynomial zeros. One step of Laguerre's iteration is

$$(8.1) \qquad L_\pm(x) = x + \cfrac{2n}{\left(\cfrac{-f'(x)}{f(x)} \pm \sqrt{(2n-1)\left((2n-1)\left(\cfrac{-f'(x)}{f(x)}\right)^2 - 2n\cfrac{f''(x)}{f(x)}\right)} \right)}.$$

For more details on the method and its properties see, e.g., [14, 20].

For a polynomial having all real roots the method is globally convergent with a cubic convergence in a neighborhood of a simple eigenvalue. If we add $\lambda_{2n+1} = -\infty$ and $\lambda_0 = \infty$ then for $x \in (\lambda_{i+1}, \lambda_i)$ we have

$$\lambda_{i+1} < L_-(x) < x < L_+(x) < \lambda_i.$$

In the divide-and-conquer algorithm we use Laguerre's method to compute the eigenvalues $\lambda_{2n} \leq \cdots \leq \lambda_1$ of Q from the initial approximations $\widetilde{\lambda}_{2n} \leq \cdots \leq \widetilde{\lambda}_1$ that are the eigenvalues of Q_0. We know from Theorem 7.3 that $\widetilde{\lambda}_{i+1} \leq \lambda_i \leq \widetilde{\lambda}_{i-1}$ and that we can use $\widetilde{\lambda}_i$ as an initial approximation for λ_i. From $\nu(Q(\widetilde{\lambda}_i))$ we see if $\lambda_i > \widetilde{\lambda}_i$ or $\widetilde{\lambda}_i < \lambda_i$ and then use the appropriate L_+ or L_- sequence. The global convergence of Laguerre's method guarantees that we get all the eigenvalues by computing them independently one by one.

Although the convergence close to a simple eigenvalue should be cubic, we can expect very slow convergence at the beginning if $\widetilde{\lambda}_i$ is closer to λ_{i-1} or λ_{i+1} than to λ_i (see Figure 8.1).
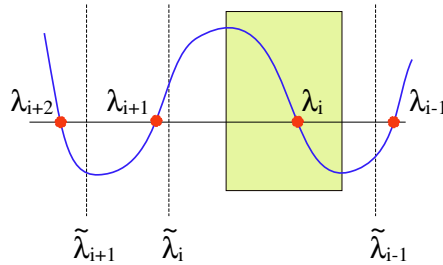
FIG. 8.1. *Slow convergence can occur when $\widetilde{\lambda}_i$ is much closer to λ_{i-1} or λ_{i+1} than to λ_i. The cubic convergence region around a simple eigenvalue λ_i.*

The necessary condition [20] for the cubic convergence near a simple eigenvalue λ is that the sign of $-f'(x)/f(x)$ agrees with the sign of $\lambda - x$ (see Figure 8.1). To improve the convergence we first use the bisection on interval $[\lambda_{i+1}, \lambda_i]$ (or $[\lambda_i, \lambda_{i-1}]$) until the condition for the cubic convergence is achieved.

Due to rounding errors, the condition $-f'(x)/f(x)(\lambda - x) > 0$ might also be achieved near λ_{i-1} or λ_{i+1}. An additional criteria that we use is that near λ_i the sign of $f'(x)$ has to agree with $(-1)^{i+1}$.

9. Ehrlich–Aberth method. This method simultaneously approximates all zeros of a polynomial $f(\lambda) = \det(Q(\lambda))$. From an initial approximation $x^{(0)} \in \mathbb{C}^{2n}$ the method generates a sequence $x^{(j)} \in \mathbb{C}^{2n}$ that locally converges to the eigenvalues of the QEP Q. The Ehrlich–Aberth iteration is given by

$$(9.1) \qquad x_j^{(k+1)} = x_j^{(k)} - \frac{\dfrac{f(x_j^{(k)})}{f'(x_j^{(k)})}}{1 - \dfrac{f(x_j^{(k)})}{f'(x_j^{(k)})} \displaystyle\sum_{\substack{l=1 \\ l \neq j}}^{2n} \frac{1}{x_j^{(k)} - x_l^{(k)}}}$$

for $j = 1, \ldots, 2n$. For details on the method and its properties see, e.g., [3, 4].

If the method is implemented in the Gauss–Seidel style then the convergence for simple roots is cubical and linear for multiple roots. We iterate only those eigenvalues that have not converged yet.

As in the previous section, we use the Ehrlich–Aberth method to compute the eigenvalues of Q using the eigenvalues of Q_0 as initial approximations. It may happen that Q_0 has multiple eigenvalues. In such case we have a division by zero in equation (9.1). In IEEE arithmetic this leads to ∞ in the denominator and consequently to $x_j^{(k+1)} = x_j^{(k)}$. To prevent this, we always slightly perturb the eigenvalues of Q_0 before we use them as initial approximations.

10. Numerical examples. We implemented Laguerre's method and the Ehrlich–Aberth method for the computation of the eigenvalues of the tridiagonal QEPs in Fortran 95. The code can be downloaded from author's web site[1]. Using Compaq Visual Fortran 6.6 on PC Pentium 4 2.6 GHz 1 GB RAM we tested both methods on a limited set of tridiagonal QEPs. In the numerical examples we compare the average number of iterations, the computational time and the accuracy of the computed eigenvalues. As a measure of the accuracy we use the maximum relative error

$$\max_{i=1,\ldots,2n} \frac{|\widetilde{\lambda}_i - \lambda_i|}{|\lambda_i|},$$

where λ_i is the exact eigenvalue computed either analytically or using variable precision in Mathematica 5. For all tridiagonal QEPs in this section we tested all three algorithms for the evaluation of the derivative of the determinant. As the choice has almost no effect on the accuracy and the number of needed steps, we include only the results for the fastest method, the three-term recurrences.

For comparison we also applied the Lapack [1] routine ZGGEV to the linearized generalized eigenvalue problem

$$(10.1) \qquad \begin{bmatrix} 0 & K \\ K & C \end{bmatrix} z = \lambda \begin{bmatrix} K & 0 \\ 0 & -M \end{bmatrix} z.$$

---

[1]http://www-lp.fmf.uni-lj.si/plestenjak/papers.htm

The main stopping criteria is the relative size of a correction. We take $\varepsilon = 10^{-15}$ and stop the iteration for λ_j when

$$|\lambda_j^{(k+1)} - \lambda_j^{(k)}| \leq |\lambda_j^{(k+1)}|\varepsilon.$$

Another stopping criteria for Laguerre's method are different inertias of $Q(\lambda_j^{(k+1)})$ and $Q(\lambda_j^{(k)})$. In the Ehrlich-Aberth method we use a heuristic that stops the iteration once the large majority of the eigenvalues has converged and the corrections for the remaining eigenvalues stop becoming smaller.

In both methods one step (an iteration for one eigenvalue approximation) has linear time complexity. If we compare the number of operations needed for the equations (8.1) and (9.1), and for the three-term recurrences in Section 4, then we can observe that one step of Laguerre's method is more expensive and is roughly equivalent to 1.8 Ehrlich–Aberth steps.

EXAMPLE 10.1. In the first numerical example we use random tridiagonal matrices, where the elements are uniformly distributed in such intervals that the obtained QEP is hyperbolic. For the matrices M and K, the diagonal and codiagonal elements are uniformly distributed in $[0.5, 1]$ and $[0, 0.1]$, respectively. The diagonal and codiagonal elements of the matrix C are uniformly distributed in $[4, 5]$ and $[0, 0.5]$, respectively.

n	Ehrlich-Aberth $\mathbb{R}$	Ehrlich-Abert $\mathbb{C}$	Laguerre-bisection	ZGGEV
average number of iterations in the last D&C				
100	1.88	1.89	1.86	
200	1.76	1.76	2.09	
400	1.57	1.58	1.20	
800	1.55	1.54	1.26	
time in seconds				
100	0.02	0.03	0.02	0.60
200	0.05	0.13	0.06	5.02
400	0.13	0.39	0.23	52.95
800	0.48	1.48	0.83	684.63
maximum relative error				
100	5e-15	4e-16	5e-15	5e-14
200	5e-15	4e-16	5e-15	9e-14
400	5e-15	4e-16	5e-15	1e-13
800	5e-15	4e-16	5e-15	1e-13

TABLE 10.1
The average number of iterations in the last divide-and-conquer step, the computational time, and the maximum relative error of the computed eigenvalues in Example 10.1.

The numerical results are presented in Table 10.1. In the first two columns are the results for the Ehrlich–Aberth method; in the first column we use real arithmetic while in the second column we use complex perturbations and complex arithmetic. Complex perturbations increase the computational time for one iteration but in some cases (see, e.g., Example 10.2), where we have multiple or close eigenvalues, we might have faster convergence. In the third column are the results for Laguerre's method and in the last column are the results for the Lapack routine ZGGEV applied to the linearized generalized eigenvalue problem (10.1) of size $2n$. The cost of ZGGEV, which is not optimized for block tridiagonal matrices, is $\mathcal{O}(n^3)$ compared to $\mathcal{O}(n^2)$ for the methods presented in this paper. Because of that ZGGEV is slower from the presented methods even for a moderate size of matrices.

We tested the methods on matrix dimensions from 100 to 800. The results in Table 10.1 are organized in three parts. In the upper part is the average number of iterations in the last divide-and-conquer step. For Laguerre's method we count bisection steps as well. As the dimension

of the matrices increases, better the eigenvalues of $Q_0(\lambda)$ approximate the eigenvalues of $Q(\lambda)$ and fewer iterations are needed in the final phase. The middle part in Table 10.1 contains the computational times in seconds. One can see that although Laguerre's method needs fewer iterations, it runs slower than the Ehrlich–Aberth method which does not compute the second derivatives. In the lower part of the table are the maximum relative errors of the computed eigenvalues. In this example all methods perform well and give small relative errors. The maximum condition number of the eigenvalues (for the definition, see, e.g., [19]) in Example 10.1 is of order 10^3.

EXAMPLE 10.2. In this example we use matrices with constant diagonals and codiagonals, such that the QEP is hyperbolic. We take $M = \mathrm{tridiag}(0.1, 1, 0.1)$, $C = \mathrm{tridiag}(0.5, 5, 0.5)$, and $K = \mathrm{tridiag}(0.2, 1, 0.2)$. For such problem the eigenvalues can be computed analytically. All eigenvalues are simple, but we can expect problems in the divide-and-conquer approach because the eigenvalues of Q_0 appear in pairs. The eigenvalues are not very sensitive, the maximum condition number for the eigenvalues in this example is of order 10^2.

Numerical results, organized in the same way as in Example 10.1, are presented in Table 10.2. We can see that the number of iterations is larger than in Example 10.1. The Ehrlich–Aberth method has problems with close initial approximations. In this case Laguerre's method gives the best performance.

n	Ehrlich-Aberth $\mathbb{R}$	Ehrlich-Abert $\mathbb{C}$	Laguerre-bisection	ZZGEV
	average number of iterations in the last D&C			
100	19.93	18.49	5.75	
200	19.57	17.55	5.70	
400	18.98	17.05	5.67	
800	18.46	16.50	5.64	
	time in seconds			
100	0.08	0.22	0.03	0.64
200	0.33	0.92	0.13	5.33
400	1.30	3.70	0.53	58.42
800	5.17	14.59	2.09	865.13
	maximum relative error			
100	3e-15	5e-16	3e-15	6e-15
200	3e-15	5e-16	3e-15	5e-15
400	3e-15	6e-16	3e-15	2e-14
800	3e-15	6e-16	3e-15	3e-15

TABLE 10.2

The average number of iterations in the last divide-and-conquer step, the computational time, and the maximum relative error of the computed eigenvalues in Example 10.2.

EXAMPLE 10.3. The Ehrlich–Aberth method in complex arithmetic can also be applied to the QEPs that are not hyperbolic and where the eigenvalues might be complex. The interlacing property of the eigenvalues of Q_0 and Q is no longer true, but we can still expect that the eigenvalues of Q_0 are good initial approximations to the eigenvalues of Q. When the solutions are complex, Laguerre's method is not globally convergent anymore and without the inertia and the interlacing property we have no guarantee that the method returns all the eigenvalues.

For the first nonhyperbolic QEP we use random symmetric tridiagonal matrices. The diagonal elements of matrices M, C and K are uniformly distributed in $[0, 1]$. The codiagonal elements of matrices M, C, and K are uniformly distributed in $[0, 0.1]$, $[0, 0.5]$, and $[0, 0.2]$, respectively. The maximum condition number of the eigenvalues is of order 10^9 and this reflects in larger errors than in the other examples.

For the second nonhyperbolic QEP we use an example from [19], where $M = \mathrm{tridiag}(0.1, 1, 0.1)$,

$C = \text{tridiag}(-3, 9, -3)$, and $K = \text{tridiag}(-5, 15, -5)$. All eigenvalues are simple, but the eigenvalues of Q_0 are double. The maximum condition number of the eigenvalues is of order 10^3.

Numerical results in Table 10.3 show that the Ehrlich–Aberth can be applied to such QEPs.

	QEP 1					QEP 2				
	ZZGEV		Ehrlich–Aberth $\mathbb{C}$			ZZGEV		Ehrlich–Aberth $\mathbb{C}$		
n	time	error	time	avg. iter	error	time	error	time	avg. iter	error
100	0.75	4e-13	0.03	1.94	1e-13	0.59	7e-15	0.23	20.10	2e-15
200	6.16	3e-12	0.11	1.72	9e-15	5.23	4e-14	1.02	19.52	2e-15
400	67.09	4e-12	0.39	1.56	4e-14	46.64	6e-14	4.09	18.88	4e-15

TABLE 10.3

The average number of iterations in the last divide-and-conquer step, the computational time, and the maximum relative error of the computed eigenvalues in Example 10.3.

EXAMPLE 10.4. We consider the second-order model of vibration of a rotating axel in a power plant from [2]. We have a second-order differential equation

$$M\ddot{z} + C\dot{z} + Kz = Du,$$

where M, C, and K are tridiagonal symmetric matrices of dimension $n = 211$. The norms of the matrices are $\|M\| = 1.7 \cdot 10^3$, $\|C\| = 2 \cdot 10^2$, and $\|K\| = 1.5 \cdot 10^{12}$. One eigenvalue of the corresponding QEP is 0, which makes the resulting system neither observable nor detectable. The largest real part of the remaining nonzero eigenvalues is $\rho = -0.01626718$.

If we apply the Ehrlich–Aberth method, then the relative error of the computed ρ is of order 10^{-14}. The average number of iterations in the last divide-and-conquer step is 6.8. If we use the linearization (10.1) and ZGGEV then the relative error of the computed ρ is of order 10^{-6}. If we reduce the linearized 422×422 problem into a 421×421 problem for the nonzero eigenvalues as in [2] then the relative error of the computed ρ rises to 10^{-9}. This example shows that we can get more accurate results without a linearization. The eigenvalues in this example have condition numbers of orders from 10^4 up to 10^{10}.

EXAMPLE 10.5. The above ideas can be extended to QEPs with banded matrices as well. We can apply the Ehrlich-Aberth method as long as we have an efficient method for the computation of the characteristic polynomial and its derivative. For banded matrices these values can be computed in a linear time using the algorithm based on the LU factorization from section 6.

As in the previous examples, the initial approximations are obtained by the divide-and-conquer scheme. The matrices M, C, and K are represented as 2×2 block matrices and then the approximations are obtained by a recursive application of the method to the diagonal block subproblems.

p	$n = 50$	$n = 100$	$n = 200$
1	3.91	2.85	2.32
2	5.78	4.18	3.31
3	6.23	5.31	4.44
4	6.42	5.91	5.33
5	9.27	6.44	6.37

TABLE 10.4

The average number of iterations in the last divide-and-conquer step for the banded quadratic eigenvalue problem with matrices of dimension n and bandwidth p from Example 10.5.

The following example was done in Matlab 7.0. We take three matrices of dimension n with normally distributed elements: M=randn(n), C=randn(n), and K=randn(n), set $m_{ij} = c_{ij} =$

$k_{ij} = 0$ for $|i - j| > p$, where p is the bandwidth, and apply the Matlab implementation of the Ehrlich-Aberth method.

As expected, the results in Table 10.5 show that the average number of iterations in the last divide-and-conquer step does increase with the bandwidth. However, for a small bandwidth, one step is performed in linear time and the results in Table 10.5 show that the Ehrlich-Aberth method can be considered as an alternative for the banded quadratic eigenvalue problems. For all combinations of p and n in Table 10.5 the maximum relative error of the computed eigenvalues is below 10^{-14} and smaller from the error obtained by the Matlab function `polyeig` that applies QZ to the linearized problem.

11. Conclusions. We have presented two numerical methods for the tridiagonal hyperbolic QEP that use the divide-and-conquer approach. Both methods can be easily parallelized. Laguerre's method and bisection require hyperbolicity, while the Ehrlich–Aberth method might be applied to more general problems, for instance, nonsymmetric tridiagonal quadratic eigenvalue problems, tridiagonal polynomial eigenvalue problems, banded polynomial eigenvalue problems, and others. In these applications, the algorithm based on the LU factorization might be used for an efficient computation of the derivative of the determinant.

Let us mention that at the moment there are no methods for transforming a general QEP to a tridiagonal form. In future, this might change with structure preserving transformations (SPT) [8].

REFERENCES

[1] E. ANDERSON ET AL., *LAPACK user's guide. Third edition*, SIAM, Philadelphia, 1999.

[2] J. ABELS AND P. BENNER, *DAREX - a collection of benchmark examples for discrete-time algebraic Riccati equations (version 2.0)*, SLICOT Working Note 1999-16, 1999.

[3] D. A. BINI, *Numerical computation of polynomial zeros by means of Aberth's method*, Numerical algorithms **13**, (1996), pp. 179–200.

[4] D. A. BINI, L. GEMIGNANI, AND F. TISSEUR, *The Ehrlich–Aberth method for the nonsymmetric tridiagonal eigenvalue problem*, SIAM J. Matrix Anal. Appl., **27** (2005), pp. 153–175.

[5] Z. BOHTE, *Calculation of the derivative of the determinant*, Obzornik mat. fiz. **28**, (1981), pp. 33–50.

[6] I. R. DHILLON, *Reliable computation of the condition number of a tridiagonal matrix in $\mathcal{O}(n)$ time*, SIAM J. Matrix Anal., **19** (1998), pp. 776–796.

[7] R. J. DUFFIN, *A minimax theory for overdamped networks*, J. Rational Mech. Anal., **4** (1955), pp. 221–233.

[8] S. D. GARVEY, *Structure preserving transformations for linear dynamic systems*, available at http://www.nottingham.ac.uk/~eazsg/SPT/index.htm.

[9] G. H. GOLUB, C. F. VAN LOAN, *Matrix computations*, 3rd ed., Johns Hopkins University Press, Baltimore, 1996.

[10] D. KRESSNER, personal communication (2003).

[11] P. LANCASTER, *Quadratic eigenvalue problems*, Linear Algebra Appl., **150** (1991), pp. 499–506.

[12] K. LI, *Durand–Kerner root-finding method for the generalized tridiagonal eigenproblem*, Missouri J. Math. Soc., (1999), pp. 33–43.

[13] K. LI AND T. Y. LI, *An algorithm for symmetric tridiagonal eigenproblems: Divide and conquer with homotopy continuation*, SIAM J. Sci. Comput., **14** (1993), pp. 735–751.

[14] T. Y. LI AND Z. ZENG, *Laguerre's iteration in solving the symmetric tridiagonal eigenproblem - revisited*, SIAM J. Sci. Comput., **15** (1994), pp. 1145–1173.

[15] K. LI, T. Y. LI, AND Z. ZENG, *An algorithm for generalized symmetric tridiagonal eigenvalue problems*, Numerical Algorithms, **8** (1994), pp. 269–291.

[16] A. S. MARKUS, *Introduction to the spectral theory of polynomial operator pencils*, Amer. Math. Soc., Providence, 1988.

[17] V. PEREYRA AND G. SCHERER, *Eigenvalues of symmetric tridiagonal matrices: a fast, accurate and reliable algorithm*, J. Inst. Math. Applics., **12** (1973), pp. 209–222.

[18] D. SAPAGOVENE, *The Sturm sequence for a nonlinear algebraic eigenvalue problem*, Upravlenija I sekcija VYP, **16** (1976), pp. 87–95.

[19] F. TISSEUR AND K. MEERBERGEN, *The quadratic eigenvalue problem*, SIAM Review, **43** (2001), pp. 235–286.

[20] J. WILKINSON, *Algebraic eigenvalue problem*, Oxford University Press, Oxford, 1965.