Numerical Methods for the Tridiagonal Hyperbolic Quadratic Eigenvalue Problem

Article *in* SIAM Journal on Matrix Analysis and Applications · January 2006

DOI: 10.1137/050624856 · Source: DBLP

1 author:

Bor Plestenjak
University of Ljubljana
36 PUBLICATIONS **262** CITATIONS

University of Ljubljana
Institute of Mathematics, Physics and Mechanics
Department of Mathematics
Jadranska 19, 1 111 Ljubljana, Slovenia

Preprint series, Vol. 43 (2005), 967

NUMERICAL METHODS FOR THE TRIDIAGONAL HYPERBOLIC QUADRATIC EIGENVALUE PROBLEM

Bor Plestenjak

ISSN 1318-4865

March 4, 2005

Ljubljana, March 4, 2005

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, the Ehrlich–Aberth method, and the Durand–Kerner method. Initial approximations are provided by a divide-and-conquer approach using rank two modifications. The above methods need a stable and efficient computation of the characteristic polynomial and its derivatives. We discuss how to obtain these values using the three-term recurrences, the QR factorization, and the LU factorization.

Key words. Quadratic eigenvalue problem, inertia, Laguerre's method, Ehrlich–Aberth method, Durand–Kerner 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)

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

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 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 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 [15].

We say that a QEP is *hyperbolic* [8] if M is positive definite and

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

for all $x \neq 0$. In this paper we focus on the real tridiagonal hyperbolic QEP, where matrices M, C, and K are symmetric and tridiagonal. 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 [4]. 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 repeat 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, 9, and 10 Laguerre's method, the Ehrlich–Aberth method, and the Durand–Kerner method, are applied to the computation of the zeros of the characteristic polynomial, respectively. Some numerical examples are given in Section 11, followed by conclusions.

*Version: February 21, 2005.

†Department of Mathematics, University of Ljubljana, Jadranska 19, SI-1000 Ljubljana, Slovenia (bor.plestenjak@fmf.uni-lj.si). The research was supported in part by the Ministry of Higher Education, Science and Technology of Slovenia (Research Project Z1-3136).

2. Auxiliary results. The following properties of the hyperbolic QEPs are gathered from [5, 8, 13]. 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 [5]). *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 [13]). *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 ν, ζ, 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, ξ, π) is the inertia of the matrix $Q(\sigma)$ then ξ 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

$$\mu_n(\lambda) \leq \cdots \leq \mu_1(\lambda), \tag{3.1}$$

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). □

REMARK 3.2. *Theorem 3.1 is a generalization of a similar theorem in [14], where M is a positive definite matrix and K is a negative definite matrix.*

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 [11, 12], the Ehrlich–Aberth method [2], and the Durand–Kerner method [9, 10].

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 the three-term recurrences, see, e.g., [6]. If $f_k(\lambda) = \det(Q_k(\lambda))$, where $Q_k(\lambda)$ is the leading $k \times k$ submatrix of $Q(\lambda)$, then

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

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 [10], 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 $f_n = d_1 \cdots d_n$,

$$\begin{array}{rclrcl} d_1 &=& a_1, & & & \\ g_0 &=& 0, & g_1 &=& \dfrac{a_1'}{a_1}, \\ h_0 &=& 0, & h_1 &=& \dfrac{a_1''}{a_1}, \end{array}$$

and

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

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 matrices M, C, and K. A similar approach is used in [11].

5. A QR factorization approach. One can see that

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

This formula is applied in [2] for a stable $\mathcal{O}(n)$ computation of $f'(\lambda)/f(\lambda)$ via QR factorization when $f(\lambda) = \det(A - \lambda I)$ and A is an unreduced tridiagonal matrix. We give a sketch of the algorithm, for details and the theory, see [2].

Let $A - \lambda I = QR$, 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 Q is the product of $n-1$ Givens rotations, $Q^* = G_{n-1}\cdots G_2G_1$, where

$$G_i([i, i+1], [i, i+1]) = \begin{bmatrix} \psi_i & \theta_i \\ -\overline{\theta_i} & \overline{\psi_i} \end{bmatrix}.$$

Then

$$Q^* = \begin{bmatrix} v_1u_1 & \psi_1 & & & 0 \\ v_2u_1 & v_2u_2 & \psi_2 & & \\ \vdots & \vdots & \ddots & \ddots & \\ \vdots & & & v_{n-1}u_{n-1} & \psi_{n-1} \\ v_nu_1 & v_nu_2 & \cdots & v_nu_{n-1} & v_nu_n \end{bmatrix},$$

where

$$D = \operatorname{diag}(1, -\psi_1, \psi_1\psi_2, \ldots, (-1)^{n-1}\psi_1\psi_2\cdots\psi_{n-1}),$$

$$u = D^{-1}[1, \overline{\theta_1}, \cdots, \overline{\theta_{n-1}}]^T,$$
$$v = D[\theta_1, \cdots, \theta_{n-1}, 1]^T.$$

If we solve $Rw = v$, then

$$\operatorname{tr}((A - \lambda I)^{-1}) = \sum_{i=1}^{n} u_i w_i. \tag{5.1}$$

Kressner [7] generalized the above approach into an $\mathcal{O}(n)$ algorithm for the computation of $\operatorname{tr}(A^{-1}B)$, 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 $\operatorname{tr}(A^{-1}B)$ we need the diagonal elements of $A^{-1}B$. From

$$\begin{aligned}(A^{-1}B)_{ii} &= e_i^T R^{-1} Q^* B e_i \\ &= z_{i-1} e_i^T R^{-1} Q^* e_{i-1} + x_i e_i^T R^{-1} Q^* e_i + y_i e_i^T R^{-1} Q^* e_{i+1},\end{aligned}$$

and

$$\begin{aligned} e_i^T R^{-1} Q^* e_{i-1} &= u_{i-1} w_i, \\ e_i^T R^{-1} Q^* e_i \quad &= u_i w_i, \\ e_i^T R^{-1} Q^* e_{i+1} &= u_{i+1} w_i + \frac{1}{r_i}(\psi_i - v_i u_{i+1}) \end{aligned}$$

it follows that

$$\operatorname{tr}(A^{-1}B) = \sum_{i=2}^{n} z_{i-1} u_{i-1} w_i + \sum_{i=1}^{n} x_i u_i w_i + \sum_{i=1}^{n-1} y_i \left(u_{i+1} w_i + \frac{1}{r_i}(\psi_i - v_i u_{i+1}) \right). \tag{5.2}$$

As reported in [2], formula (5.1) is not stable. To make it stable, we have to avoid the explicit multiplication by matrix D or D^{-1}. If we define $\hat{R} = D^{-1} R D$, $\hat{v} = D^{-1} v$, $\hat{u} = Du$, and solve $\hat{R}\hat{w} = \hat{v}$ for $\hat{w}$, then

$$\operatorname{tr}((A - \lambda I)^{-1}) = \sum_{i=1}^{n} \hat{u}_i \hat{w}_i. \tag{5.3}$$

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

Using the same notation it follows from

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

that we may rewrite formula (5.2) in a stable form

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

6. A LU factorization approach. In [3] 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 exist analytic matrices $L(\mu)$ and $U(\mu)$ such that

$$L(\mu)U(\mu) = PQ(\mu) \tag{6.1}$$

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 [3]). *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.*

for $r = 1$ to n
  for $k = r$ to n
    $v_{rk} = b_{rk} - \sum_{j=1}^{r-1} (m_{rj} u_{jk} + l_{rj} v_{jk})$
  for $i = r + 1$ to n

$$m_{ir} = \frac{1}{u_{rr}} \left(b_{ir} - \sum_{j=1}^{r-1} (m_{ij} u_{jr} + l_{ij} v_{jr}) - l_{ir} v_{rr} \right)$$

For the second derivative we have

$$PQ'' = L''U + 2L'U' + LU'' = NU + 2MV + LW, \tag{6.2}$$

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 the 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 [3] 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

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

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 $\widetilde{\lambda}_{2n} \le \cdots \le \widetilde{\lambda}_1$ of Q_0, a union of the eigenvalues of Q_1 and Q_2, are approximations to the eigenvalues $\lambda_{2n} \le \cdots \le \lambda_1$ 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

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

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. For each vector $x' \in \mathbb{R}^m$ we can define $x \in \mathbb{R}^n$ as $x = \begin{bmatrix} x'^T & 0 \end{bmatrix}^T$. Since

$$x'^T Q_1(\gamma) x' = x^T Q(\gamma) x,$$

matrix $Q_1(\gamma)$ (and similarly $Q_2(\gamma)$) is negative definite. For each nonzero $x \in \mathbb{R}^n$, partitioned as $x = [x_1^T \; x_2^T]^T$, where $x_1 \in \mathbb{R}^m$ and $x_2 \in \mathbb{R}^{n-m}$, we have

$$x^T Q_0(\gamma) x = x_1^T Q_1(\gamma) x_1 + x_2^T Q_2(\gamma) x_2 < 0,$$

and Q_0 is hyperbolic because of Theorem 2.2. One can see from

$$x^T Q_t(\gamma) x = (1 - t) x^T Q_0(\gamma) x + t x^T Q(\gamma) x < 0$$

that Q_t is a hyperbolic QEP for $t \in [0,1]$. □

The following theory is a generalization of Theorem 5.2 in [12].

LEMMA 7.2. *Let $\lambda_{2n}(t) \le \cdots \le \lambda_1(t)$ 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 $i = 1, \ldots, 2n$.*

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 order $2n$. 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 $t \in [0,1]$. □

THEOREM 7.3. *Let $\widetilde{\lambda}_{2n} \le \cdots \le \widetilde{\lambda}_1$ be the eigenvalues of $Q_0(\lambda)$ and $\lambda_{2n} \le \cdots \le \lambda_1$ the eigenvalues of Q. Then:*

a) $\widetilde{\lambda}_1 \le \lambda_1$ and $\lambda_{2n} \le \widetilde{\lambda}_{2n}$,
b) $\widetilde{\lambda}_{i+1} \le \lambda_i \le \widetilde{\lambda}_{i-1}$ for $i = 2, \ldots, n-1$ and $i = n+2, \ldots, 2n-1$,
c) $\widetilde{\lambda}_{n+1} \le \lambda_{n+1} < \lambda_n \le \widetilde{\lambda}_n$.

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. □

REMARK 7.4. *On the contrary to 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 $\widetilde{\lambda}_1, \ldots, \widetilde{\lambda}_{2n}$. Three numerical methods that may be applied for this task are presented in the next three sections.

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

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

For more details on the method and its properties see, e.g., [11, 16].

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 $\lambda_{2n} \le \cdots \le \lambda_1$ of Q from the initial approximations $\widetilde{\lambda}_{2n} \le \cdots \le \widetilde{\lambda}_1$ that are the eigenvalues of Q_0. We know from Theorem 7.3 that $\widetilde{\lambda}_{i+1} \le \lambda_i \le \widetilde{\lambda}_{i-1}$ 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 in the beginning if $\widetilde{\lambda}_i$ is closer to λ_{i-1} or λ_{i+1} than to λ_i (see Figure 8.1).

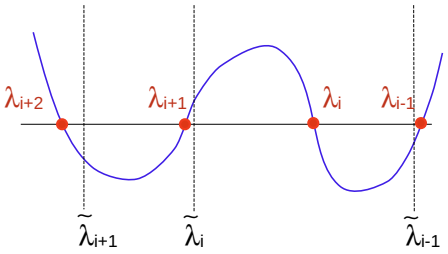

FIG. 8.1. *Slow convergence can occur when $\widetilde{\lambda}_i$ is much closer to λ_{i-1} or λ_{i+1} than to λ_i.*

The necessary condition [16] 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 8.2). 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.

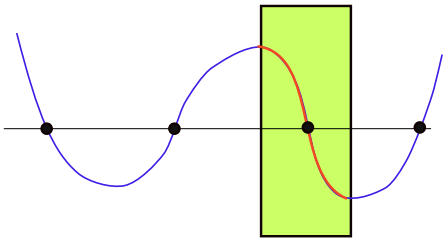

FIG. 8.2. *The cubic convergence region around a simple eigenvalue.*

Due to the 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

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

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

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. Durand–Kerner method. This is another method that simultaneously approximates all zeros of a polynomial. As the method requires that the leading coefficient of the polynomial is one, we apply it on

$$p(\lambda) = \frac{1}{\det(M)} \det(Q(\lambda)).$$

Similar to the Ehrlich–Aberth method we generate a sequence $x^{(j)} \in \mathbb{C}^{2n}$ that locally converges to the eigenvalues of $Q(\lambda)$. The equation for the Durand–Kerner method is

$$x_j^{(k+1)} = x_j^{(k)} - \frac{p(x_j^{(k)})}{\displaystyle\prod_{\substack{l=1 \\ l\neq j}}^{2n} (x_j^{(k)} - x_l^{(k)})} \tag{10.1}$$

for $j = 1, \ldots, 2n$. For details on the method and its properties see, e.g., [9].

If we implement the method in the Gauss–Seidel style then the convergence for simple roots is superquadratical and linear for multiple roots. As in the previous section, we iterate only the eigenvalues that have not converged yet.

As in the Ehrlich–Aberth method, a division by zero occurs in (10.1) when Q_0 has multiple eigenvalues. To prevent this, we always slightly perturb the eigenvalues of Q_0 before we use them as initial approximations.

11. Numerical examples. In Matlab 7.0 on PC Pentium4 2.6 GHz 1 GB RAM we tested all three methods (Laguerre's method, the Ehrlich–Aberth method, and the Durand–Kerner method) 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 in Mathematica 5 using variable precision and $\widetilde{\lambda}_i$ is an approximation computed in Matlab. Numerical experiments (see Example 11.2) show that the choice of the method for the computation of the derivatives of the characteristic polynomial has almost no effect on the accuracy and the number of needed steps. Unless it is mentioned otherwise, the three-term recurrences are used for the computation of $\det(Q(\lambda))$.

In all three 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), (9.1), and (10.1), together with the number of operations needed for the three-term recurrences in Section 4, then we can observe that one step of the Durand–Kerner method is the cheapest as it needs only values of f. One step of the Ehrlich–Aberth method needs f and f' and is roughly equivalent to 2 Durand–Kerner steps. One step of Laguerre's method is even more expensive. It requires f, f', and f'', and is roughly equivalent to 3.7 Durand–Kerner steps.

EXAMPLE 11.1. 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 matrices M and K, diagonal and codiagonal elements are uniformly distributed in $[0.5, 1]$ and $[0, 0.1]$, respectively. Diagonal and codiagonal elements of matrix C are uniformly distributed in $[4, 5]$ and $[0, 0.5]$, respectively.

n	Ehrlich-Aberth $\mathbb{R}$	Durand-Kerner $\mathbb{R}$	Ehrlich-Aberth $\mathbb{C}$	Durand-Kerner $\mathbb{C}$	Laguerre-bisection	`polyeig`
average number of iterations in the last D&C						
50	2.46	4.91	2.52	5.29	2.19	
100	2.15	5.88	2.26	6.61	1.71	
200	1.99	13.23	2.17	9.84	1.91	
400	1.86	4.08	2.04	4.59	1.16	
800	1.85	4.05	2.03	4.72	1.22	
time in seconds						
50	0.27	0.41	0.36	0.73	0.36	0.13
100	0.53	0.98	0.92	2.38	0.78	0.53
200	1.19	3.27	2.58	9.20	2.33	5.19
400	2.78	6.38	7.66	23.91	4.08	52.34
800	7.09	15.55	25.48	75.55	9.91	929.78
maximum relative error						
50	2e-16	2e-16	2e-16	2e-16	7e-15	5e-15
100	2e-16	4e-16	4e-16	2e-16	6e-15	1e-14
200	2e-16	7e-13	2e-16	3e-16	4e-15	1e-14
400	3e-16	3e-16	2e-16	3e-16	2e-15	2e-14
800	2e-16	3e-16	3e-16	3e-16	5e-15	2e-14

TABLE 11.1

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 11.1.

The numerical results are presented in Table 11.1. In the first four columns are the results for the Ehrlich–Aberth and the Durand–Kerner method. In the first two columns we use real arithmetic, while in the third and the fourth column we use complex perturbations and complex arithmetic. Complex perturbations increase the computational time for one iteration but in some cases (see Examples 11.3 and 11.4), where we have multiple or close eigenvalues, we might have faster convergence. In the fifth column are the results for Laguerre's method and in the last column are the results for the Matlab function `polyeig`, which uses linearization and is not optimized for tridiagonal matrices. One can see that `polyeig` is slower from the presented methods even for a moderate size of matrices.

We tested the methods on matrix dimensions from 50 to 800. The results in Table 11.1 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 11.1 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. The Durand–Kerner method is the fastest per one iteration but as it converges slower than the other two methods it needs more time. 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.

EXAMPLE 11.2. In this example we compare three methods for the computation of the derivative of the determinant from Sections 4, 5, and 6. We use the Ehrlich–Aberth method and the same set of test matrices as in Example 11.1.

n	Three-term recurrences	QR factorization	LU factorization
average number of iterations in the last D&C			
50	2.46	2.47	2.45
100	2.15	2.12	2.15
200	1.99	2.01	2.00
400	1.86	1.88	1.87
800	1.85	1.85	1.82
time in seconds			
50	0.25	0.38	0.33
100	0.50	0.84	0.78
200	1.17	2.08	1.95
400	2.75	5.23	5.08
800	6.83	14.13	14.47
maximum relative error			
50	2.5e-16	2.3e-16	2.2e-16
100	2.2e-16	2.2e-16	3.8e-16
200	2.4e-16	2.8e-16	2.1e-16
400	2.7e-16	2.7e-16	2.7e-16
800	2.3e-16	2.5e-16	3.0e-16

TABLE 11.2

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 11.2.

The numerical results in Table 11.2 show that there are only minor differences in the number of the needed steps and in the accuracy of the computed eigenvalues. The methods differ only in the computational time. Since the method that uses the three-term recurrences is the fastest, we use only this method in the remaining numerical examples.

EXAMPLE 11.3. In this example we use matrices with constant diagonals and codiagonals, such that the QEP is hyperbolic. We take $M = \text{tridiag}(0.1, 1, 0.1)$, $C = \text{tridiag}(0.5, 5, 0.5)$, and $K = \text{tridiag}(0.2, 1, 0.2)$. 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.

Numerical results, organized in the same way as in Example 11.1, are presented in Table 11.3. We can see that the number of iterations is larger than in Example 11.1. The Ehrlich–Aberth method and especially the Durand–Kerner method have problems with close initial approximations. In this case Laguerre's method gives the best performance. The Durand–Kerner method converges slowly and gives large relative errors. This is probably because the equation (10.1) is very sensitive for close approximations.

n	Ehrlich-Aberth $\mathbb{R}$	Durand-Kerner $\mathbb{R}$	Ehrlich-Aberth $\mathbb{C}$	Durand-Kerner $\mathbb{C}$	Laguerre-bisection	`polyeig`
average number of iterations in the last D&C						
50	13.55	23.95	10.66	18.39	4.32	
100	12.91	84.77	10.17	23.76	4.34	
200	12.38	> 250	9.51	19.32	4.32	
400	11.87	> 250	9.20	21.56	4.30	
800	11.25	> 250	8.63	21.57	4.29	
time in seconds						
50	0.56	0.77	0.80	1.39	0.41	0.16
100	1.48	4.38	2.52	5.69	0.95	0.61
200	3.83	> 34	8.30	19.00	2.33	5.50
400	10.25	> 133	28.89	76.78	5.86	58.88
800	30.03	> 509	99.64	311.11	15.59	1140.19
maximum relative error						
50	4e-16	3e-15	4e-16	4e-16	3e-15	5e-15
100	5e-16	5e-04	5e-16	5e-16	3e-15	4e-15
200	5e-16	6e-02	5e-16	4e-16	5e-15	5e-15
400	5e-16	9e-01	5e-16	5e-16	5e-15	5e-15
800	5e-16	5e-00	5e-16	7e-07	6e-15	7e-15

TABLE 11.3

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 11.3.

EXAMPLE 11.4. We use

$$Q(\lambda) = \begin{bmatrix} Q_1(\lambda) & A(\lambda) & \\ A(\lambda)^T & Q_1(\lambda) & A(\lambda) \\ & A(\lambda)^T & Q_1(\lambda) \end{bmatrix},$$

where Q_1 is the QEP from Example 11.1 and $A(\lambda) = 10^{-4}\lambda e_n e_1^T$. The eigenvalues appear in clusters of order three.

Numerical results, organized in the same way as in the previous examples, are presented in Table 11.4. Similar to the previous example, the performance of the Durand–Kerner method is not satisfactory, while the Ehrlich–Aberth method and Laguerre's method perform well. We believe that by using the modified Laguerre method and by estimating the multiplicities of the eigenvalues as in [11], we could further improve the performance of Laguerre's method on QEPs with clusters of eigenvalues.

EXAMPLE 11.5. The Ehrlich–Aberth and the Durand–Kerner method in complex arithmetic can also be applied of the methods 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.

Numerical results in Table 11.5 show that the Ehrlich–Aberth and the Durand–Kerner method perform well and can be applied to such QEPs.

EXAMPLE 11.6. For the second nonhyperbolic QEP we use an example from [15], where $M = \text{tridiag}(0.1, 1, 0.1)$, $C = \text{tridiag}(-3, 9, -3)$, and $K = \text{tridiag}(-5, 15, -5)$. All eigenvalues

n	Ehrlich-Aberth $\mathbb{R}$	Durand-Kerner $\mathbb{R}$	Ehrlich-Aberth $\mathbb{C}$	Durand-Kerner $\mathbb{C}$	Laguerre-bisection	`polyeig`
average number of iterations in the last D&C						
51	6.00	96.01	10.13	29.77	13.91	
102	7.64	52.92	16.10	37.67	12.52	
201	8.71	52.18	17.53	39.29	11.69	
402	8.42	> 250	17.87	39.94	8.53	
time in seconds						
51	0.30	2.48	0.59	1.75	0.52	0.14
102	0.69	3.59	2.33	6.75	1.14	0.63
201	1.75	9.92	8.22	24.17	2.77	5.16
402	4.31	> 100	30.19	89.75	7.28	51.91
maximum relative error						
51	1e-15	1e-07	7e-16	1e-15	3e-14	5e-15
102	2e-14	6e-08	8e-16	2e-15	1e-13	9e-15
201	2e-14	1e-08	9e-16	2e-15	1e-13	2e-14
402	2e-14	5e-07	1e-15	2e-15	1e-13	2e-14

TABLE 11.4

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 11.4.

	`polyeig`		Ehrlich–Aberth $\mathbb{C}$			Durand–Kerner $\mathbb{C}$		
n	time	error	time	avg. iter	error	time	avg. iter	error
50	0.14	1e-14	0.52	2.29	9e-16	0.83	4.72	9e-16
100	0.89	5e-14	1.42	2.32	8e-15	2.69	5.87	2e-15
200	7.14	9e-14	2.97	2.15	6e-15	7.67	6.20	1e-14
400	68.44	2e-13	8.97	2.09	3e-15	73.55	3.98	9e-15

TABLE 11.5

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 11.5.

are simple, but the eigenvalues of Q_0 are double. The results imply that the Ehrlich–Aberth and the Durand–Kerner method can be applied to nonhyperbolic QEPs as well.

	`polyeig`		Ehrlich–Aberth $\mathbb{C}$			Durand–Kerner $\mathbb{C}$		
n	time	error	time	avg. iter	error	time	avg. iter	error
50	0.11	8e-15	0.95	13.54	4e-15	1.80	23.45	8e-16
100	0.69	1e-14	3.09	12.88	9e-16	7.44	31.40	2e-15
200	5.70	1e-14	10.56	12.69	5e-15	26.23	27.27	3e-15
400	49.23	3e-14	36.77	12.06	4e-15	111.11	34.36	3e-15

TABLE 11.6

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 11.6.

12. Conclusions. We have presented three numerical methods for the tridiagonal hyperbolic QEP. All methods can be easily parallelized. The divide-and-conquer approach combined with the Ehrlich–Aberth or Durand–Kerner 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.

Acknowledgements. The author would like to thank Daniel Kressner for providing the generalization of the QR factorization approach in Section 5. The author would also like to thank Dario Bini and Françoise Tisseur for useful suggestions and comments.

REFERENCES

[1] D. A. Bini, *Numerical computation of polynomial zeros by means of Aberth's method*, Numerical algorithms **13**, (1996), pp. 179–200.
[2] D. A. Bini, L. Gemignani, and F. Tisseur, *The Ehrlich–Aberth method for the nonsymmetric tridiagonal eigenvalue problem*, NA Report 428, University of Manchester, (2003).
[3] Z. Bohte, *Calculation of the derivative of the determinant*, Obzornik mat. fiz. **28**, (1981), pp. 33–50.
[4] 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.
[5] R. J. Duffin, *A minimax theory for overdamped networks*, J. Rational Mech. Anal., **4** (1955), pp. 221–233.
[6] G. H. Golub, C. F. van Loan, *Matrix computations*, 3rd ed., Johns Hopkins University Press, Baltimore, 1996.
[7] D. Kressner, personal communication (2003).
[8] P. Lancaster, *Quadratic eigenvalue problems*, Linear Algebra Appl., **150** (1991), pp. 499–506.
[9] K. Li, *Durand–Kerner root-finding method for the generalized tridiagonal eigenproblem*, Missouri J. Math. Soc., (1999), pp. 33–43.
[10] 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.
[11] 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.
[12] K. Li, T. Y. Li, and Z. Zeng, *An algorithm for generalized symmetric tridiagonal eigenvalue problems*, Numerical Algorithms, **8** (1994), pp. 269–291.
[13] A. S. Markus, *Introduction to the spectral theory of polynomial operator pencils*, Amer. Math. Soc., Providence, 1988.
[14] D. Sapagovene, *The Sturm sequence for a nonlinear algebraic eigenvalue problem*, Upravlenija I sekcija VYP, **16** (1976), pp. 87–95.
[15] F. Tisseur and K. Meerbergen, *The quadratic eigenvalue problem*, SIAM Review, **43** (2001), pp. 235–286.
[16] J. Wilkinson, *Algebraic eigenvalue problem*, Oxford University Press, Oxford, 1965.