On solving the definite tridiagonal symmetric generalized eigenvalue problem

Raf Vandebril
Gene Golub
Marc Van Barel

Report TW485, January 2007

Katholieke Universiteit Leuven
Department of Computer Science
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

On solving the definite tridiagonal symmetric generalized eigenvalue problem

Raf Vandebril
Gene Golub
Marc Van Barel

Report TW485, January 2007

Department of Computer Science, K.U.Leuven

Abstract

In this manuscript we will present a new fast technique for solving the generalized eigenvalue problem $T\mathbf{x} = \lambda S\mathbf{x}$, in which both matrices T and S are symmetric tridiagonal matrices and the matrix S is assumed to be positive definite.[1]

A method for computing the eigenvalues is translating it to a standard eigenvalue problem of the following form: $L^{-1}TL^{-T}(L^T\mathbf{x}) = \lambda(L^T\mathbf{x})$, where $S = LL^T$ is the Cholesky decomposition of the matrix S.

We will prove in this manuscript, that the matrix $L^{-1}TL^{-T}$, with $S = LL^T$ is of structured rank form. More precisely the matrix is of quasiseparable form, meaning that all submatrices taken out of the strictly lower triangular part of the matrix, have rank at most 1. These matrices admit an order $O(n)$ representation, instead of $O(n^2)$ in case the matrix was dense. It will be shown that the computation of the generators of this quasiseparable matrix is only linear in time.

Exploiting the properties of structured rank matrices one can use different techniques for computing all of the eigenvalues. Either one can reduce the quasiseparable matrix $L^{-1}TL^{-T}$ to tridiagonal form in $O(n^2)$ operations, and hence compute the eigenvalues of the tridiagonal matrix. One can also immediately compute the eigenvalues of this matrix, via the QR-algorithm for quasiseparable matrices. Both approaches lead to $O(n^2)$ methods for computing all the eigenvalues of the initial generalized eigenvalue problem, instead of the $O(n^3)$ methods in case the structure was left unexploited. Moreover for quasiseparable matrices also bisection and divide and conquer methods exist.

Keywords : generalized eigenvalue problem, quasiseparable, divide and conquer algorithm for quasiseparable, reduction to tridiagonal
AMS(MOS) Classification : Primary : 65F15, Secondary : 15A18.

On solving the definite tridiagonal symmetric generalized eigenvalue problem*

Raf Vandebril †, Gene Golub‡, Marc Van Barel §

8th January 2007

Abstract

In this manuscript we will present a new fast technique for solving the generalized eigenvalue problem $T\mathbf{x} = \lambda S\mathbf{x}$, in which both matrices T and S are symmetric tridiagonal matrices and the matrix S is assumed to be positive definite.[1]

A method for computing the eigenvalues is translating it to a standard eigenvalue problem of the following form:

$$L^{-1}TL^{-T}(L^T\mathbf{x}) = \lambda(L^T\mathbf{x}),$$

where $S = LL^T$ is the Cholesky decomposition of the matrix S.

We will prove in this manuscript, that the matrix $L^{-1}TL^{-T}$, with $S = LL^T$ is of structured rank form. More precisely the matrix is of quasiseparable form, meaning that all submatrices taken out of the strictly lower triangular part of the matrix, have rank at most 1. These matrices admit an order $O(n)$ representation, instead of $O(n^2)$ in case the matrix was dense. It will be shown that the computation of the generators of this quasiseparable matrix is only linear in time.

Exploiting the properties of structured rank matrices one can use different techniques for computing all of the eigenvalues. Either one can reduce the quasiseparable matrix $L^{-1}TL^{-T}$ to tridiagonal form in $O(n^2)$ operations, and hence compute the eigenvalues of the tridiagonal matrix. One can also immediately compute the eigenvalues of this matrix, via the QR-algorithm for quasiseparable matrices. Both approaches lead to $O(n^2)$ methods for computing all the eigenvalues of the initial generalized eigenvalue problem, instead of the $O(n^3)$ methods in case the structure was left unexploited. Moreover for quasiseparable matrices also bisection and divide and conquer methods exist.

Keywords: generalized eigenvalue problem, quasiseparable, divide and conquer algorithm for quasiseparable, reduction to tridiagonal

1 Introduction

In this paper we will consider the generalized eigenvalue problem of the following form:

$$T\mathbf{x} = \lambda S\mathbf{x},$$

*The research was partially supported by the Research Council K.U.Leuven, project OT/05/40 (Large rank structured matrix computations), Center of Excellence: Optimization in Engineering, by the Fund for Scientific Research–Flanders (Belgium), Iterative methods in numerical Linear Algebra), G.0455.0 (RHPH: Riemann-Hilbert problems, random matrices and Padé-Hermite approximation), G.0423.05 (RAM: Rational modelling: optimal conditioning and stable algorithms), and by the Belgian Programme on Interuniversity Poles of Attraction, initiated by the Belgian State, Prime Minister's Office for Science, Technology and Culture, project IUAP V-22 (Dynamical Systems and Control: Computation, Identification & Modelling). The first author has a grant of "Postdoctoraal Onderzoeker" from the Funds of Scientific Research Flanders (FWO-Vlaanderen). The scientific responsibility rests with the authors.

†Dept. Computerwetenschappen, K.U.Leuven, Belgium. raf.vandebril@cs.kuleuven.be
‡Dept. Computer Science, Stanford University, USA. golub@sccm.stanford.edu
§Dept. Computerwetenschappen, K.U.Leuven, Belgium. marc.vanbarel@cs.kuleuven.be

[1] In case S is negative definite, we change the problem to $-T\mathbf{x} = \lambda(-S)\mathbf{x}$, such that $-S$ is positive definite.

where both the matrices T and S are symmetric tridiagonal matrices and the matrix S is considered positive definite. This problem arises in several applications such as the numerical solution of the radial Schrödinger and Sturm-Liouville equations and vibrational analysis [1, 2]. More references to applications can be found in [3].

A method for solving this problem is the reduction to a standard eigenvalue problem in the following sense:

$$L^{-1}TL^{-T}(L^T\mathbf{x}) = \lambda(L^T\mathbf{x}),$$

where $S = LL^T$, is the Cholesky decomposition of the matrix S (see [4, 5]). This approach is considered as inefficient because the generated matrix $L^{-1}TL^{-T}$ is assumed to be dense and the accuracy of the method is dependent of the condition of S as the inverse of its Cholesky factors is required. This method requires $O(n^3)$ operations, when computing the eigenvalues as described. Also different techniques exist, such as the one presented in [3] by computing the eigenvalues via the associated characteristic polynomial, via Laguerre's iteration. There exist divide and conquer methods [6, 7]. Also methods for band matrices exist [8]. More references can be found in [3].

In this manuscript we will prove that the considered matrix $L^{-1}TL^{-T}$ is dense, but has underlying low rank properties. More precisely the matrix will be of quasiseparable form [9, 10]. This means that all submatrices taken out of the strictly lower triangular part of this matrix will be of rank at most 1.

Based on the theoretical proof that the considered matrix is quasiseparable, we will present a methods for effectively computing the representation of the quasiseparable matrix. As the quasiseparable matrix is highly structured only $O(n)$ parameters are needed to characterize the complete matrix. We will represent the quasiseparable matrix using the Givens-vector representation [11]. A fast $O(n)$ method for transforming the generalized eigenvalue problem towards an eigenvalue problem involving a quasiseparable matrix will be given.

Based on the $O(n)$ representation of the quasiseparable matrix, alternative, much faster methods for computing the whole spectrum will be considered. Different methods exist for computing the eigenvalues of quasiseparable matrices. One can easily reduce the quasiseparable matrix with $O(n^2)$ operations to tridiagonal form, instead of $O(n^3)$ for a dense matrix [12, 13, 14]. Then one can simply compute the eigenvalues of this tridiagonal matrix. Or one can, instead of reducing the matrix to tridiagonal form and computing its eigenvalues, immediately apply an implicit/explicit QR-algorithm on the considered quasiseparable matrix [15, 16]. Also other techniques such as bisection and Sturm sequence methods [17] and divide and conquer methods exist for quasiseparable matrices [18, 19]. [2]

The manuscript is organized as follows. In the first section some definitions and a theoretical proof of the structure of the matrix $L^{-1}TL^{-T}$ are given. To conclude this section a description is given on the representation used for the quasiseparable matrix and a method for effectively computing the quasiseparable matrix representation. Section 3 briefly discusses some well-known methods for computing the eigenvalues of quasiseparable matrices. The final section of this manuscript presents some numerical experiments showing thereby the speed and the accuracy of the new technique for computing the eigenvalues of the generalized eigenvalue problem.

2 Transforming the generalized eigenvalue problem

In this section we will prove theoretically that the considered matrix is dense, but not dense in its representation. In other words, we will prove that all matrices taken out of the part below the diagonal will have rank at most equal to 1. Let us first formally define what is meant with a quasiseparable matrix.

Definition 1. *A matrix $A \in \mathbb{R}^{n\times n}$ is named a quasiseparable matrix (of quasiseparability rank 1) if any submatrix taken out of the strictly lower triangular part has rank at most 1 (a similar demand holds for the upper triangular part). More precisely this means that for every $i = 2, \ldots, n$[3]:*

$$\operatorname{rank} A(i:n, 1:i-1) \leq 1.$$

[2]Some of the references above deal with semiseparable plus diagonal matrices instead of quasiseparable matrices. The techniques presented in these references can however be adapted easily to be suitable for quasiseparable matrices.

[3]We use MATLAB-style notation.

In the remainder of the text we will also need lower/upper triangular semiseparable matrices. Let us define them.

Definition 2. *A matrix $A \in \mathbb{R}^{n\times n}$ is named a semiseparable matrix (of semiseparability rank* 1*) if any submatrix taken out of the lower triangular part has rank at most* 1 *(a similar demand holds for the upper triangular part). More precisely this means that for every $i = 1,\ldots,n$*

$$\operatorname{rank} A(i:n, 1:i) \leq 1.$$

The only difference between quasiseparable and semiseparable is the fact that a quasiseparable matrix does not have the diagonal included in the low rank structure, whereas a semiseparable matrix does have this diagonal included in the structure.

2.1 Theoretical proof of the structure

Reconsidering now the matrix product $L^{-1}TL^{-T}$, with L satisfying $S = LL^T$. Due to the fact that the matrix S is tridiagonal, the matrix L will be of lower bidiagonal form and the matrix $L^{-1}TL^{-T}$ will be of quasiseparable form.

We will formulate this as a theorem.

Theorem 1. *Suppose a symmetric tridiagonal matrix T is given, and L is a lower bidiagonal matrix. The matrix*

$$A = L^{-1}TL^{-T},$$

will be a symmetric quasiseparable matrix.

Proof. We assume the considered matrices to be of size n.

It is well known that the inverse of a lower bidiagonal matrix is a lower triangular semiseparable matrix (see e.g. [20]). Let us denote $M = L^{-1}$.

The QR-factorization of such a lower semiseparable matrix can easily be computed by performing an upgoing sequence of Givens transformations, $n-1$ in total. More precisely the first Givens transformation is performed on the bottom two rows of the matrix M, to remove the complete last row up to the diagonal. Remark that it is possible to remove this complete row with one transformation as the last and second last row are dependent of each other due to the semiseparable structure. The second last Givens transformation acts on row $n-2$ and $n-1$, and removes the whole row $n-1$ up to the diagonal. This procedure can easily be repeated and gives us the following factorization:

$$G_1^T G_2^T \ldots G_{n-2}^T G_{n-1}^T M = R,$$

with R an upper triangular matrix. This means that $M = G_{n-1}G_{n-2}\ldots G_2G_1R$, which is the QR-factorization of the considered matrix.

Let us now look closer at the structure of the matrix

$$L^{-1}T = MT = G_{n-1}G_{n-2}\ldots G_2G_1RT.$$

An easy calculation reveals that the matrix RT is an upper Hessenberg matrix. We will now indicate with figures what happens with this upper Hessenberg matrix if we apply our Givens transformations G_1 up to G_{n-1} onto this matrix. We show this on a 5×5 example, as this illustrates the general case. In the following figures the elements $\times$ denote arbitrary elements, whereas the elements $\boxtimes$ denote elements belonging to the quasiseparable rank 1 part. Our Hessenberg matrix is of the following form:

$$\begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & & \times & \times & \times \\ & & & \times & \times \end{bmatrix}.$$

Applying the first transformation, G_1 onto this matrix does not change its structure as the Givens transformation acts only on the first two rows. Applying the second transformation G_2 changes the second and the third row. Due to the zero element in row 3, this element becomes dependent of the element above and we obtain:

$$\begin{bmatrix} \times & \times & \times & \times & \times \\ \boxtimes & \times & \times & \times & \times \\ \boxtimes & \times & \times & \times & \times \\ & & \times & \times & \times \\ & & & \times & \times \end{bmatrix}.$$

The third transformation places a multiple of the first two elements of row three in row four and gives us:

$$\begin{bmatrix} \times & \times & \times & \times & \times \\ \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times \\ & & & \times & \times \end{bmatrix}.$$

Finally the last transformation places a multiple of the first three elements of row four in the first three elements of row five.

$$\begin{bmatrix} \times & \times & \times & \times & \times \\ \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times \end{bmatrix} = \begin{bmatrix} \times & \times & \times & \times & \times \\ \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times \end{bmatrix}.$$

In the figure on the right we included one more element in the quasiseparable structure as this does not have an impact on the global low rank structure. We can clearly see that the lower triangular part of this matrix is of quasiseparable form. This means that the matrix $L^{-1}T$ has a lower quasiseparable structure.

Due to the fact that the matrix L^{-T} is upper triangular, it is obvious that a multiplication of $L^{-1}T$ on the right with the matrix L^{-T} does not change the low rank structure below the diagonal. Hence we have proved that our matrix $A = L^{-1}TL^{-T}$ has the lower triangular part of quasiseparable form. Due to symmetry also the upper triangular part satisfies these constraints and hence the complete matrix is quasiseparable. $\square$

Let us now see how we can effectively represent a quasiseparable matrix and how to compute this representation.

2.2 A representation for this matrix

We proved in the previous theorem that the resulting matrix is of quasiseparable form. To be able to work with the matrix an effective representation of this low rank part is necessary. A straightforward choice might be to represent the low rank part as coming from a rank 1 matrix. This means, representing the lower triangular part as coming from $\mathbf{u}\mathbf{v}^T$, with $\mathbf{u}$ and $\mathbf{v}$ two vectors. This is however a bad choice. First of all, this representation does not cover all kinds of quasiseparable matrices and second of all it suffers heavily from numerical instabilities, when computing e.g. the spectrum via a QR-method for quasiseparable matrices. More information on the problems with this representation can be found in [11]. There exist various kinds of other suitable representations, such as the quasiseparable, diagonal-subdiagonal representation, Givens-vector representation, ... In this manuscript we will focus on the Givens-vector representation. Let us briefly recapitulate some of the results for this representation, before showing how we can effectively compute it for the considered quasiseparable matrix.

To represent the strictly lower triangular part of the quasiseparable matrix, we will use a representation consisting of $n-2$ Givens transformations and a vector of length $n-1$. The Givens transformations are denoted as $G = [G_1, \ldots, G_{n-2}]$ and the vector as $\mathbf{v} = [v_1, \ldots, v_{n-1}]$. It is clearly seen that this representation needs only $O(n)$ parameters. The diagonal of the quasiseparable matrix is stored separately, leading to a global storage of $2n-1$ elements and $n-2$ Givens transformations.

The following figures denote how the strictly lower triangular part of the matrix can be reconstructed. We only show here the strictly lower triangular part of the quasiseparable matrix. The elements denoted by $\boxtimes$ make up the low rank part of the matrix. Initially one starts on the first 2 rows of the matrix. The element v_1 is placed in the upper left position, then a Givens transformation is applied, and finally to complete the first step element v_2 is added in position $(2,1)$. Only the first two columns and rows are shown here.

$$\begin{bmatrix} v_1 & 0 \\ 0 & 0 \end{bmatrix} \rightarrow G_1 \begin{bmatrix} v_1 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & v_2 \end{bmatrix} \rightarrow \begin{bmatrix} \boxtimes & 0 \\ \boxtimes & v_2 \end{bmatrix}.$$

The second step consists of applying the Givens transformation G_2 on the second and the third row, furthermore v_3 is added in position $(3,3)$. Here only the first three columns are shown and the second and third row. This leads to:

$$\begin{bmatrix} \boxtimes & v_2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \rightarrow G_2 \begin{bmatrix} \boxtimes & v_2 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & v_3 \end{bmatrix} \rightarrow \begin{bmatrix} \boxtimes & \boxtimes & 0 \\ \boxtimes & \boxtimes & v_3 \end{bmatrix}.$$

This process can be repeated by applying the Givens transformation G_3 on the third and the fourth row of the matrix, and afterwards adding the diagonal element v_4. After applying all the Givens transformations and adding all the diagonal elements, the strictly lower triangular part of a quasiseparable matrix is constructed. Because of the symmetry also the strictly upper triangular part is known. Finally one obtains a strictly lower triangular part of the following form.

$$\begin{bmatrix} c_1 v_1 & & & \\ c_2 s_1 v_1 & c_2 v_2 & & \\ c_3 s_2 s_1 v_1 & c_3 s_2 v_2 & c_3 v_3 & \\ \vdots & \vdots & & \ddots \end{bmatrix} \tag{1}$$

We remark that the construction of the quasiseparable part of the matrix, resembles the application of the Givens transformations onto the Hessenberg matrix in the proof of Theorem 1.

2.3 Computing the representation

Let us now show how we can easily compute the generators of this quasiseparable matrix. We have to compute three things to obtain the generators of the resulting quasiseparable matrix: The Givens transformations for the Givens-vector representation, the vector used in this representation and the diagonal elements of the resulting quasiseparable matrix.

Let us start by computing effectively the Givens transformations. Suppose we have the lower bidiagonal matrix L. It is obvious that there exists a sequence of Givens transformations $G_{n-1}\ldots G_1$ applied on the right of the matrix L such that $LG_{n-1}\ldots G_1 = \hat{L}$ is an upper bidiagonal matrix. The Givens transformation G_{n-1} works on the last two columns, the transformation G_{n-2} on columns $n-2$ and $n-1$ and so forth. This gives us the following equation:

$$\begin{aligned} L^{-1}TL^{-T} &= \left(\hat{L}G_1^T G_2^T \ldots G_{n-1}^T\right)^{-1} TL^{-T} \\ &= G_{n-1}\ldots G_2 G_1 \hat{L}^{-1} TL^{-T} \\ &= G_{n-1}\ldots G_2 \left(G_1 \hat{L}^{-1} TL^{-T}\right). \end{aligned}$$

Combining the factors $G_1\hat{L}^{-1}TL^{-T} = H$, we get an upper Hessenberg matrix. Hence it is clear due to construction that the Givens transformations G_{n-1} up to G_2 are the Givens transformations needed for the Givens-vector representation of the quasiseparable part in the matrix. Moreover we see that it only takes $O(n)$ operations to compute all these Givens transformations, more precisely only $8n-8$ operations are involved.

In the remainder of this section we will illustrate how to compute the vector $\mathbf{v}$ of the Givens-vector representation of the strictly lower triangular part, and also how to compute the diagonal of the resulting quasiseparable matrix. To construct these remaining unknown generators, we have to compute in some sense explicitly the product of these matrices. Fortunately due to all the hidden structure, the global cost of computing all these generators is $O(n)$. Not all the details of the computational aspects are given as it often involves straightforward computations. The software is available in MATLAB, and can be downloaded from the first author's site. The following list gives a brief overview of the consecutive steps which will be followed to compute the diagonal and the vector $\mathbf{v}$. After each item the number of involved operations is shown, indicating the low cost of computing the generators of the quasiseparable matrix. Following this list we provide more detailed comments on the computations.

- Compute the generators of the inverse of L. *Computing the Givens transformations involves* $8n-8$ *operations, the vector takes* $2n-1$ *operations.*[4]
- Compute the representation of the matrix TL^{-T}, subdivided in:
  - compute the diagonal elements *(*$6n-6$ *operations)*;
  - compute the subdiagonal elements *(*$2n-2$ *operations)*;
  - compute the generators of the strictly upper triangular part *(*$12n-21$ *operations)*.
- Compute the representation of the matrix $L^{-1}TL^{-T}$, subdivided in:
  - compute the generators of the strictly lower triangular part *(*$16n-32$ *operations)*;
  - compute the diagonal elements *(*$10n-14$ *operations)*.

Let us start by computing the generators of the inverse of L. The matrix L is lower bidiagonal, hence its inverse is a lower semiseparable matrix. Therefore, this matrix can be represented by a Givens-vector representation. We do already have the Givens transformations for the representation, of the matrix L^{-1}, namely $G_1, G_2, \ldots G_{n-1}$. As the diagonal elements of L^{-1} are the inverses of the diagonal elements of L, one has as generators the Givens from above and as vector elements the following vector:

$$\mathbf{v}_L = \left[\frac{1}{l_{11}c_1}, \frac{1}{l_{22}c_2}, \ldots, \frac{1}{l_{n-1,n-1}c_{n-1}}, \frac{1}{l_{nn}} \right].$$

Assuming that we denote our matrices as $L = (l_{ij})_{ij}$. We remark that these computations are well-defined as one can easily verify that all cosines in the different Givens transformations are different from zero.[5]

Following the computation of the inverse we compute the matrix TL^{-T}. With similar techniques as discussed in the proof one can see that this matrix is an upper Hessenberg matrix, for which the strictly upper triangular part is of quasiseparable form. In fact one can compute by simple matrix multiplication the diagonal and subdiagonal elements of the matrix TL^{-T}. The computation of the generators of the upper triangular part is also rather straightforward. The Givens transformations are exactly the same as the one used above, only one less: $G_2, G_3, \ldots, G_{n-1}$. The vector of the representation of this upper triangular part can be obtained by computing the superdiagonal elements of the matrix TL^{-T} and dividing them by the corresponding cosines of the Givens transformations.

Finally, we need to compute the diagonal and strictly lower triangular representation of the quasiseparable matrix $L^{-1}(TL^{-T})$. Even though it seems that we will obtain a dense matrix the multiplication between the lower semiseparable matrix L^{-1} and the strictly upper triangular part of (TL^{-T}), which is of quasiseparable form, can be done in $O(n)$ operations. Writing down the lower semiseparable matrix, and the strictly upper triangular part of the matrix TL^{-T} as in Equation 1, one can easily deduce a simple loop which computes the subdiagonal elements of the new quasiseparable matrix. Based on these subdiagonal elements and on the fact that the cosines of the Givens transformations are different from zero, one can easily obtain the generators of the strictly lower triangular part.

Summarizing, the complexity of computing the Cholesky decomposition of the positive definite tridiagonal matrix takes $5n-3$ operations. Computing the generators of the quasiseparable matrix takes $56n-84$

[4] An operation consists of performing one of the following operations $+, -, \times, /$.

[5] A cosine equal to zero, translates to the fact that a diagonal element of L needed to be zero, which is not possible due to the positive definiteness of S.

operations. Hence the total cost for computing the representation of the quasiseparable matrix is $61n-87$ operations.

Traditionally, one assumed that the approach of computing the eigenvalues via $L^{-1}TL^{-T}$, was too expensive because the reduction to tridiagonal form of a dense matrix already took $O(n^3)$ operations. This reduction was essential, before being able to compute the spectrum in $O(n^2)$ operations, via for example a divide and conquer technique or a QR-algorithm. Using the method presented above however, we see that it takes $O(n)$ operations to obtain the representation of the quasiseparable matrix. For this quasiseparable matrix there exist techniques $O(n^2)$ for computing the whole spectrum. In the next section we will briefly discuss this methods.

3 Computing the eigenvalues of a quasiseparable matrix

We do not go into the details on how to compute the eigenvalues of quasiseparable matrices, as these techniques are well-known nowadays. We will just present some pointers to manuscripts in which all the essential information can be found.

3.1 Reduction to tridiagonal form

Due to the specific rank structure of the matrix A, we can easily reduce this matrix to tridiagonal form in $O(n^2)$ operations instead of the traditional reduction, which needs $O(n^3)$ operations. There exist several variants to reduce a quasiseparable matrix to tridiagonal form [13, 12]. Also a kind of a parallel method to reduce a quasiseparable matrix to tridiagonal form was developed [14]. Recently also more general reduction schemes, to reduce arbitrary structured rank matrices to tridiagonal (Hessenberg in the nonsymmetric case), were proposed [21, 22]. All presented algorithms are of order $O(rn^2)$, where r is a factor related to the rank of the structured rank parts. In our case we consider a quasiseparable matrix of quasiseparability rank $r=1$.

3.2 Applying the QR-algorithm directly onto the quasiseparable matrix

The last few years people have intensively studied QR-algorithms for structured rank matrices. Let us present some of these results. First there was the QR-algorithm for semiseparable matrices, followed by the one for semiseparable plus diagonal matrices [23, 24]. Recently more general types of QR-algorithms exist for low rank perturbations of unitary matrices and so forth [25, 26, 27]. Explicit QR-algorithms for higher order quasiseparable matrices can be found in the following manuscripts [15, 16].

3.3 Other methods

As mentioned in the introduction also different techniques for computing the eigenvalues of quasiseparable matrices exist. For example the bisection method and a method based on Sturm sequences can be found in [17].

Another technique is based on halving at every step of the algorithm the problem size. These so called divide and conquer methods are based on solving the secular equation [18, 19].

Both methods mentioned above need $O(n^2)$ operations for computing the whole spectrum.

4 Numerical experiments

In this section we will present results concerning the timings and accuracy of the presented approach. We chose to use the divide and conquer method for computing the eigenvalues and eigenvectors of the considered quasiseparable matrices. Up to our knowledge this algorithm is among the fastest available methods and provides accurate results when computing the eigenvalues of quasiseparable matrices. The software was implemented in MATLAB and executed on a Linux platform. The used divide and conquer approach is the one from [18], where we needed to adapt the software as the implementation as presented in [18] (the straightforward solver) was based on the generator representation whereas the resulting quasiseparable matrix in our approach is represented using the Givens-vector representation.

In the first set of experiments we show results concerning the speed of the complete algorithm for computing eigenvalues of the definite symmetric tridiagonal matrix pair. In the second set of experiments we show results concerning the accuracy of the computed eigenvalues and eigenvectors.

4.1 Timings

The left Figure 1 shows the speed of constructing the quasiseparable matrix, based on the two tridiagonal matrices. Both the timings of the individual experiments as well as the average cputime are shown. We performed experiments on matrices of sizes ranging from 100 up to 2000 and for each size 5 experiments were run. The timings as depicted in the left figure are divided by the size of the experiment. This clearly shows that we get a complexity of $O(n)$ for computing the representation of the quasiseparable matrix.

The figure on the right plots the cputime needed to compute the whole spectrum including the eigenvectors of the quasiseparable matrix, using the divide and conquer method. The timings are divided by the square of the problem size. It is clear that the average time graph converges to a straight line, indicating that the complexity is indeed $O(n^2)$.

Hence, the numerical experiments clearly indicate that the presented approach for computing the eigenvalues of the definite general symmetric tridiagonal eigenvalue problem is of order $O(n^2)$.

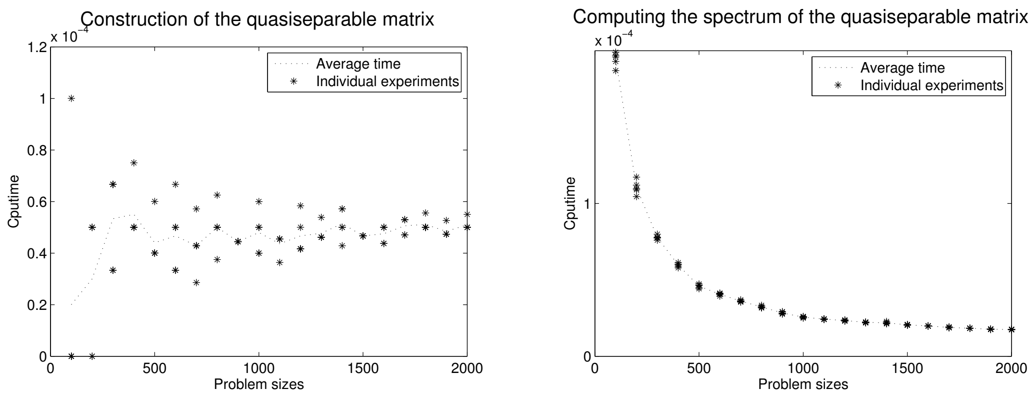

Figure 1: Timings for computing the whole spectrum

The MATLAB method EIG(T,S,'CHOL') computes the generalized eigenvalues by computing the eigenvalues of the symmetric matrix $L^{-1}TL^{-T}$, with $S = LL^T$. It reduces the symmetric problem to a tridiagonal eigenvalue problem by orthogonal similarity transformations. This reduction to tridiagonal form, does not exploit the quasiseparable structure, hence this reduction will use $O(n^3)$ operations.

4.2 Accuracy, experiment 1

In the first experiment we compared the computed eigenvalues with known eigenvalues of the problem. We solved the definite symmetric generalized eigenvalue problem with two tridiagonal Toeplitz matrices. The first matrix T is constructed with a random diagonal and a random (using RAND of MATLAB) subdiagonal(equal to the superdiagonal) element. The second matrix S has a random subdiagonal (equal to the superdiagonal) element, whereas the diagonal element is chosen to be twice the subdiagonal element plus 1. In this way we known that the matrix S is positive definite and moreover is well-conditioned.

As both Toeplitz matrices commute we can explicitly compute the spectrum of the generalized eigenvalue problem as it equals the ratios of the (ordered) eigenvalues of T and S. In the experiment the eigenvalues of T and S respectively were computed using the EIG of MATLAB. Based on these 'correct' eigenvalues of the generalized eigenvalue problem, we performed experiments for sizes ranging from 100 to 1500, and for each experiment we performed five random tests (with the constraints on S as mentioned above). The relative accuracy measure which we took into consideration is the following one:

$$\frac{\max_i |\lambda_i - \tilde{\lambda}_i|}{(\max_i \lambda_i)},$$

where λ_i denote the eigenvalues and $\tilde{\lambda}$ the computed eigenvalues.

It is clear in Figure 3 that the presented approach is very accurate and computes in average all eigenvalues up to machine precision.

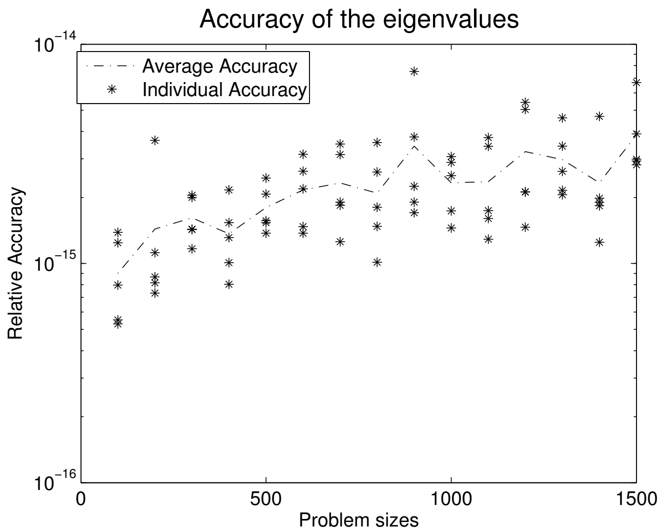

Figure 2: Accuracy of the eigenvalues

4.3 Accuracy, experiment 2

In the following set of experiments we compute a relative error involving the eigenvalues and eigenvectors. We solve in fact the first problem involving the quasiseparable matrix A of the two equivalent problems:

$$\begin{aligned} A\mathbf{y} &= \lambda \mathbf{y}, \\ L^{-1}TL^{-T}(L^T\mathbf{x}) &= \lambda(L^T\mathbf{x}). \end{aligned}$$

We compute hence the eigenvalues λ_i corresponding to the eigenvectors $\mathbf{y}_i$. To obtain the eigenvectors $\mathbf{x}_i$ of the generalized eigenvalue problem, we need to compute the following:

$$\mathbf{x}_i = L^{-T}\mathbf{y}.$$

As we know from the theoretical results, the matrix L^{-T} is an upper triangular semiseparable matrix. Moreover the representation in terms of Givens transformations and a vector is known. The multiplication between the matrix $L^{-T}\mathbf{y}$ can easily be performed in $O(n)$ operations (see e.g.[28]).

For the following set of experiments we took matrix sizes ranging from 100 to 2000 and 5 experiments for each size were considered. The tridiagonal matrix T has random diagonal and subdiagonal elements. The matrix S has random subdiagonal elements and the diagonal elements were taken $S(i,i) = 2 * \max(S(i,i-1), S(i,i+1)) + 1$, in order to make the matrix positive definite and well conditioned.

We considered the following relative backward error:

$$\max_i \left(\frac{\|T\mathbf{x}_i - \lambda_i S\mathbf{x}_i\|_2}{\|T\mathbf{x}_i\|_2 + |\lambda_i| \, \|S\mathbf{x}_i\|_2)} \right),$$

where the eigenvectors $\mathbf{x}_i = L^{-T}\mathbf{y}_i$ and the eigenvalues λ_i and eigenvectors $\mathbf{y}_i$ were computed using the presented method.

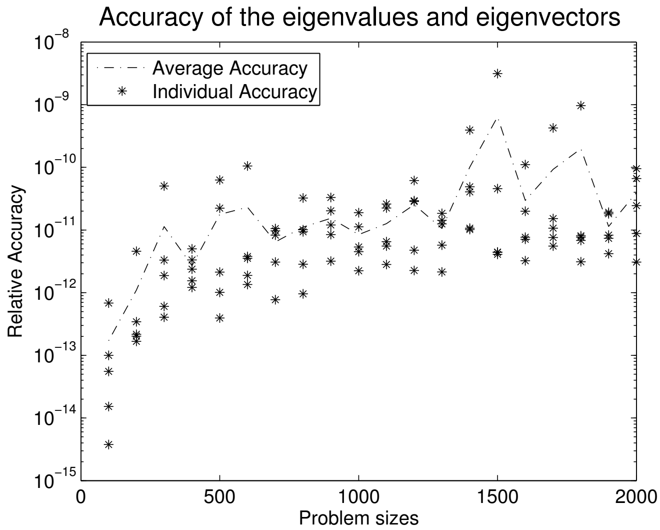

Figure 3: Accuracy of the eigenvalues/eigenvectors

4.4 Accuracy, experiment 3

In this last experiment we will perform more specific tests, for comparing the accuracy of the proposed method with the methods, available in MATLAB.

The matrices S considered in the following examples are tridiagonal matrices having eigenvalues $2^0, 2^1, \ldots, 2^{n-1}$. They are constructed by reducing the matrix:

$$Q^T \operatorname{diag}([2^0, 2^1, \ldots, 2^{n-1}])Q,$$

to tridiagonal form. The matrix Q is an orthogonal matrix taken from the QR-factorization of a random matrix. The diagonal and subdiagonal elements of the matrix T are generated by the MATLAB command RANDN. We computed the generalized eigenvalues for this problem by three methods. The method proposed in this manuscript and by the QZ-method EIG(T,S,'QZ') implemented in MATLAB and by the Cholesky factorization as implemented in MATLAB EIG(T,S,'CHOL').

We plotted for all eigenpairs the following error. Remark that we normalized the eigenvectors:

$$\|T\mathbf{x}_i - \lambda_i S\mathbf{x}_i\|_2.$$

In the left of Figure 4 you can see the comparison of the different accuracies of the different methods. The right figure denotes the logarithm of the absolute values of the eigenvalues, to see which eigenvalues are the smallest ones in the left figure. It is shown that our method performs better for smaller eigenvalues than the Cholesky factorization approach of MATLAB. It performs slightly worse than the QZ-approach, for small eigenvalues. One can also see that the new approach yields a behavior rather similar to the one of the Cholesky approach.

In the following four figures, we repeated the same experiment. A similar behavior as discussed in the previous figure is observed.

In Figure 6, we did not compare the individual error for each eigenpair, but we ran 20 experiments (like the ones above) and plotted for each experiment individually the following error measure:

$$\|TE - SE\Lambda\|_2,$$

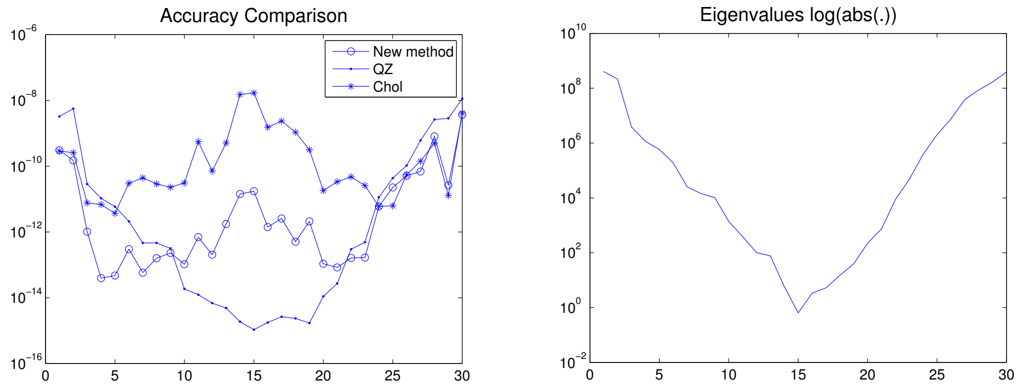

Figure 4: Accuracy of the eigenvalues/eigenvectors

the matrix E contains the eigenvectors and the matrix Λ is a diagonal matrix containing the eigenvalues on the diagonal. We can see that globally the new method delivers the most accurate results.

Even though we sort of reshuffled the eigenvalues of the original matrix by performing the similarity transformation with the matrix Q, and reducing the matrix to tridiagonal form, the matrix is still graded in some sense. The top left elements on the diagonal are much smaller than the elements in the bottom right position. We will now flip our matrix S upside down and from left to right, such that it becomes graded but in the wrong dimension.

In Figure 7 on the left, we plotted for one specific example the error of each eigenpair. We see that the Cholesky approach and the new method perform equally well. In the right figure, we plot again the global error of 20 experiments. We see that the QZ-method is the worst now, whereas the Cholesky approach is slightly better than the new method.

5 Conclusions

In this manuscript we showed that one can solve the definite generalized tridiagonal symmetric eigenvalue problem by transforming it to a standard eigenvalue problem. The presented method is can be solved efficiently in $O(n^2)$ operations instead of $O(n^3)$ by exploiting the structured rank properties of the involved coefficient matrix.

It was shown that this matrix is of quasiseparable form and its eigenvalues and eigenvectors can be computed efficiently using various methods.

Numerical experiments showed that the computational complexity is $O(n^2)$, and moreover the method provides accurate results for all test cases considered.

Acknowledgements

The authors wish to thank Françoise Tisseur for fruitful discussions at the ILAS meeting in the Netherlands 2006.

References

[1] R. F. Boisvert. Families of high order accurate discretizations of some elliptic problems. *SIAM Journal on Scientific and Statistical Computation*, 2(3):268–284, 1981.

[2] Y. M. Ram and G. M. L. Gladwell. Constructing a finite element model of a vibratory rod from eigendata. *Journal of Sound and Vibration*, 169(2):229–237, 1994.

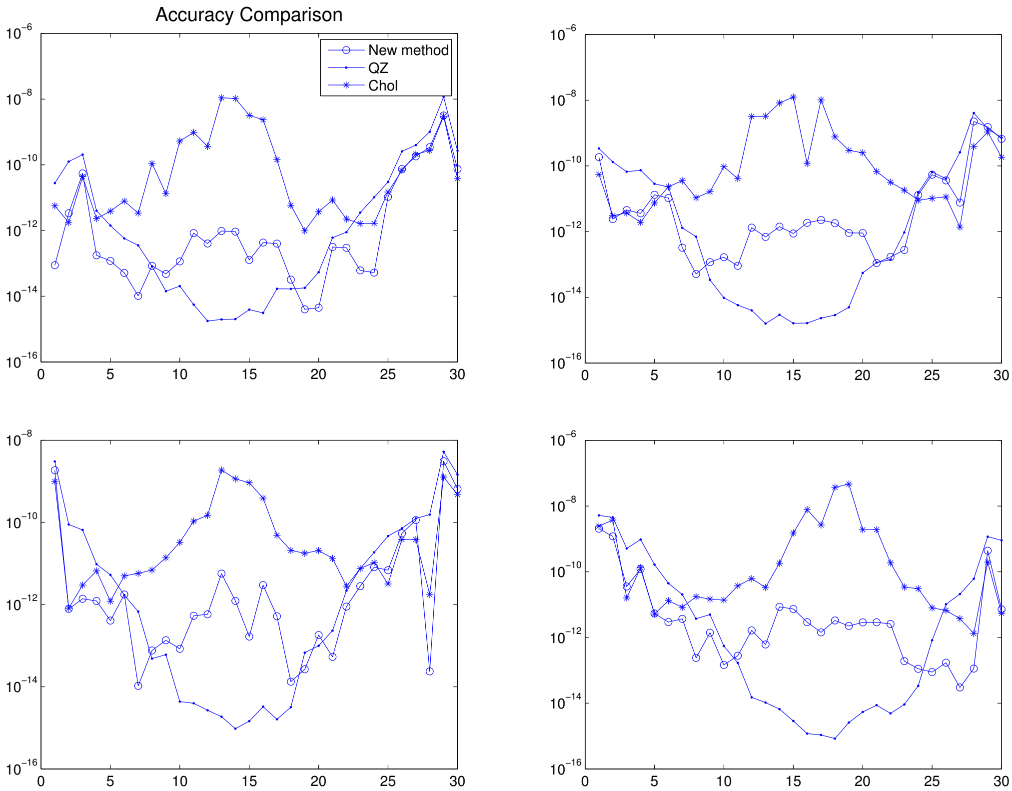

Figure 5: Accuracy of the eigenvalues/eigenvectors

[3] K. Li, T.-Y. Li, and Z. Zeng. An algorithm for the generalized symmetric tridiagonal eigenvalue problem. *Numerical Algorithms*, 8(2–4):269–291, 1994.

[4] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, third edition, 1996.

[5] B. N. Parlett. *The Symmetric Eigenvalue Problem*, volume 20 of *Classics in Applied Mathematics*. SIAM, Philadelphia, 1998.

[6] C. F. Borges and W. B. Gragg. A parallel divide and conquer algorithm for the generalized real symmetric definite tridiagonal eigenproblem. *Numerical Linear Algebra (Kent, OH, 1992)*, pages 11–29, 1993.

[7] C. F. Borges and W. B. Gragg. Divide and conquer for generalized real symmetric definite tridiagonal eigenproblems. In *Proc. Shanghai International Conference on Numerical Linear Algebra and Applications*, pages 70–76, Shanghai, October 1992.

[8] L. Kaufman. An algorithm for the banded symmetric generalized matrix eigenvalue problem. *SIAM Journal on Matrix Analysis and its Applications*, 14(2):372–389, 1993.

[9] Y. Eidelman and I. C. Gohberg. On a new class of structured matrices. *Integral Equations and Operator Theory*, 34:293–324, 1999.

[10] Y. Eidelman. Fast recursive algorithm for a class of structured matrices. *Applied Mathematics Letters*, 13:57–62, 2000.

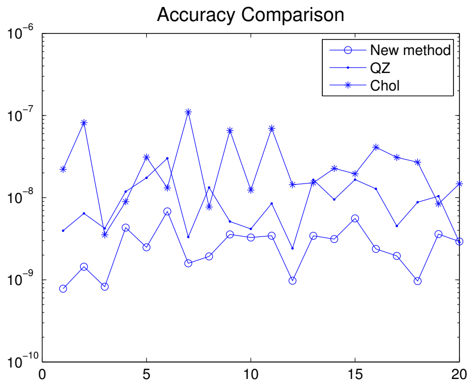

Figure 6: Global accuracy

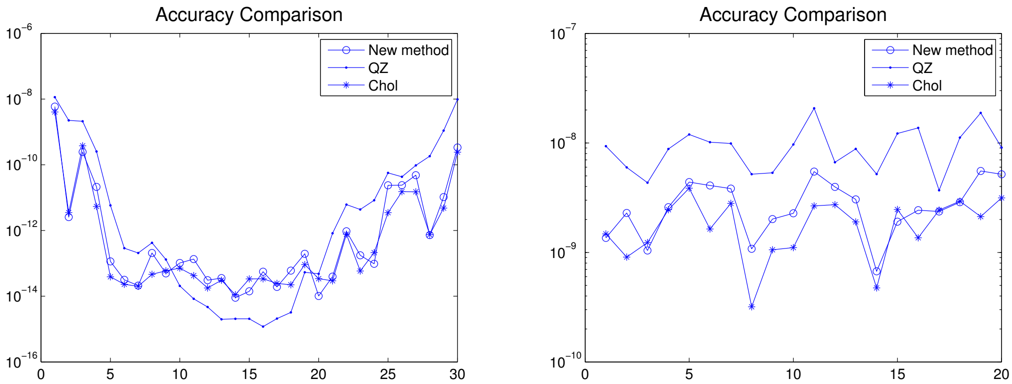

Figure 7: Individual accuracy and global accuracy

[11] R. Vandebril, M. Van Barel, and N. Mastronardi. A note on the representation and definition of semiseparable matrices. *Numerical Linear Algebra with Applications*, 12(8):839–858, October 2005.

[12] N. Mastronardi, S. Chandrasekaran, and S. Van Huffel. Fast and stable algorithms for reducing diagonal plus semiseparable matrices to tridiagonal and bidiagonal form. *BIT*, 41(1):149–157, 2003.

[13] D. Fasino, N. Mastronardi, and M. Van Barel. Fast and stable algorithms for reducing diagonal plus semiseparable matrices to tridiagonal and bidiagonal form. *Contemporary Mathematics*, 323:105–118, 2003.

[14] N. Mastronardi, S. Chandrasekaran, and S. Van Huffel. Fast and stable two-way algorithm for diagonal plus semi-separable systems of linear equations. *Numerical Linear Algebra with Applications*, 8(1):7–12, 2001.

[15] Y. Eidelman, I. C. Gohberg, and V. Olshevsky. The QR iteration method for Hermitian quasiseparable matrices of an arbitrary order. *Linear Algebra and its Applications*, 404:305–324, July 2005.

[16] S. Delvaux and M. Van Barel. The explicit QR-algorithm for rank structured matrices. Technical Report TW459, Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, 3000 Leuven (Heverlee), Belgium, May 2006.

[17] Y. Eidelman, I. C. Gohberg, and V. Olshevsky. Eigenstructure of order-one-quasiseparable matrices. three-term and two-term recurrence relations. *Linear Algebra and its Applications*, 405:1–40, 2005.

[18] N. Mastronardi, M. Van Barel, and E. Van Camp. Divide and conquer algorithms for computing the eigendecomposition of symmetric diagonal-plus-semiseparable matrices. *Numerical Algorithms*, 39(4):379–398, 2005.

[19] S. Chandrasekaran and M. Gu. A divide and conquer algorithm for the eigendecomposition of symmetric block-diagonal plus semi-separable matrices. *Numerische Mathematik*, 96(4):723–731, February 2004.

[20] M. Fiedler and T. L. Markham. Completing a matrix when certain entries of its inverse are specified. *Linear Algebra and its Applications*, 74:225–237, 1986.

[21] S. Delvaux and M. Van Barel. A Hessenberg reduction algorithm for rank structured matrices. Technical Report TW460, Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, 3000 Leuven (Heverlee), Belgium, May 2006.

[22] Y. Eidelman, L. Gemignani, and I. C. Gohberg. On the fast reduction of a quasiseparable matrix to Hessenberg and tridiagonal forms. *Linear Algebra and its Applications*, 2006. (To appear).

[23] R. Vandebril, M. Van Barel, and N. Mastronardi. An implicit *QR*-algorithm for symmetric semiseparable matrices. *Numerical Linear Algebra with Applications*, 12(7):625–658, 2005.

[24] E. Van Camp, M. Van Barel, R. Vandebril, and N. Mastronardi. An implicit *QR*-algorithm for symmetric diagonal-plus-semiseparable matrices. Technical Report TW419, Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, 3000 Leuven (Heverlee), Belgium, March 2005.

[25] D. A. Bini, F. Daddi, and L. Gemignani. On the shifted *QR* iteration applied to companion matrices. *Electronic Transactions on Numerical Analysis*, 18:137–152, 2004.

[26] D. A. Bini, Y. Eidelman, L. Gemignani, and I. C. Gohberg. Fast QR eigenvalue algorithms for Hessenberg matrices which are rank-one perturbations of unitary matrices.

[27] D. A. Bini, L. Gemignani, and V. Y. Pan. Fast and stable *QR* eigenvalue algorithms for generalized companion matrices and secular equations. *Numerische Mathematik*, 100(3):373–408, 2005.

[28] R. Vandebril. *Semiseparable matrices and the symmetric eigenvalue problem*. PhD thesis, Dept. of Computer Science, K.U.Leuven, Celestijnenlaan 200A, 3000 Leuven, May 2004.