A QZ-algorithm for semiseparable matrices

Yvette Vanberghen *Raf Vandebril*
Marc Van Barel

Report TW471, October 2006

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

A QZ-algorithm for semiseparable matrices

Yvette Vanberghen　　*Raf Vandebril*
Marc Van Barel

Report TW 471, October 2006

Department of Computer Science, K.U.Leuven

Abstract

This manuscript focusses on the translation of the traditional eigenvalue problem, based on sparse matrices, towards a structured rank approach. An effective reduction of a matrix pair to lower semiseparable, upper triangular form will be presented as well as a QZ-method for this matrix pair. Important to remark is that this reduction procedure also inherits a kind of nested subspace iteration as was the case in the regular eigenvalue problem based on semiseparable matrices. It will also be shown, that the QZ-method for structured rank matrices is closely related to the traditional QZ-method for sparse matrices.

Keywords : semiseparable, QZ-algorithm, generalized eigenvalue problem.
AMS(MOS) Classification : Primary : 65F15, Secondary : 15A18.

A *QZ*-algorithm for semiseparable matrices ⋆

Yvette Vanberghen [a], Raf Vandebril [a], Marc Van Barel [a]

[a] *K.U.Leuven, Department of Computer Science, Celestijnenlaan 200A, 3001 Heverlee (Leuven), Belgium.*

Abstract

This manuscript focusses on the translation of the traditional eigenvalue problem, based on sparse matrices, towards a structured rank approach. An effective reduction of a matrix pair to lower semiseparable, upper triangular form will be presented as well as a *QZ*-method for this matrix pair. Important to remark is that this reduction procedure also inherits a kind of nested subspace iteration as was the case in the regular eigenvalue problem based on semiseparable matrices. It will also be shown, that the *QZ*-method for structured rank matrices is closely related to the traditional *QZ*-method for sparse matrices.

Key words: semiseparable, *QZ*-algorithm, generalized eigenvalue problem

1 Introduction

Nowadays a lot of attention is paid to finding alternative approaches for solving eigenvalue and related problems, based on structured rank matrices. The first publications w.r.t. eigenvalue computations and structured rank matrices focussed on the effective computation of the eigenvalues of such matrices by reducing them to

⋆ The research was partially supported by the Research Council K.U.Leuven, OT/05/40 (Large rank structured matrix computations), Center of Excellence: Optimization in Engineering, by the Fund for Scientific Research–Flanders (Belgium), projects G.0078.01 (SMA: Structured Matrices and their Applications), G.0176.02 (ANCILA: Asymptotic aNalysis of the Convergence behavior of Iterative methods in numerical Linear Algebra), G.0184.02 (CORFU: Constructive study of Orthogonal Functions) 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 second author has a grant of "Postdoctoraal Onderzoeker" from the Fund of Scientific Research Flanders (FWO-Vlaanderen).

Email addresses: `yvette.vanberghen@cs.kuleuven.be` (Yvette Vanberghen), `raf.vandebril@cs.kuleuven.be` (Raf Vandebril), `marc.vanbarel@cs.kuleuven.be` (Marc Van Barel).

URLs: `www.cs.kuleuven.be/∼raf` (Raf Vandebril), `www.cs.kuleuven.be/∼marc` (Marc Van Barel).

sparse matrices [1,2]. There are divide and conquer approaches [3] and methods for directly performing *QR*-steps on these matrices [4,5]. Due to the strong relation between lower semiseparable and Hessenberg matrices, one started to search for possible translations of the complete eigenvalue problem from sparse matrix arithmetic to the dense structured rank approach. Several algorithms were deduced, for reducing the matrices to structured rank form [6] and for performing the *QR*-algorithms on these matrices. The new 'structured rank' approach involved slightly more operations in the reduction as well as in the *QR*-step. Surprisingly enough, this increased complexity does not necessarily lead to a global larger complexity when computing the whole or part of the spectrum. These extra involved operations create an extra convergence behavior, which can lead to deflation already in the reduction to structured rank form, leading to an immediate reduction in complexity of the following *QR*-method. A study of these results, the convergence properties inside the reduction and a global cost comparison was presented in [7].

The aim of this paper is to present an alternative method for computing the eigenvalues of a pencil of two n by n matrices A and B, based on structured rank matrices. The eigenvalues of this pencil are the elements of the set $\Lambda(A,B)$ and if $\lambda \in \Lambda(A,B)$, this means that:

$$Ax = \lambda Bx.$$

Similarly like in the regular eigenvalue problem, we have to compute these eigenvalues via an iterative procedure. A traditional procedure, as can be found for example in [8,9], first reduces, via unitary transformations the pencil $A - \lambda B$ to Hessenberg-triangular form. Following this reduction an iterative procedure, named the *QZ*-method is performed on this transformed couple, to bring both matrices to upper triangular form. The generalized eigenvalue problem is a special case of more general polynomial matrix eigenvalue problems. A nice overview of the quadratic eigenvalue problem, solution methods and applications can be found in the overview article by Meerbergen and Tisseur [10]

Here, an alternative reduction to structured rank form will be presented. Instead of a Hessenberg and an upper triangular matrix as intermediate matrices, a matrix which is lower semiseparable and an upper triangular matrix will be used. We will numerically show that we can recover the specific (advantageous) convergence behavior from the simple eigenvalue problem based on structured rank matrices.

Having a reduction algorithm is not sufficient for computing all the generalized eigenvalues. Hence we present also the associated *QZ*-algorithm. It will be shown that there is close relation between this *QZ*-method and the traditional *QZ*-method performed on a Hessenberg-triangular pair.

The manuscript is organized as follows. In section 2 an algorithm is presented to reduce a pair of two general matrices (A,B) to a lower semiseparable, upper triangular matrix pair. Section 3 describes the *QZ*-algorithm for a semiseparable-triangular matrix pair. Some numerical experiments are discussed in section 4 followed by some conclusions.

1.1 Semiseparable matrices and notation

Before continuing the analysis of the generalized eigenvalue problem, we will briefly define semiseparable matrices, and present some notations.

Definition 1 *A matrix $A \in \mathbb{R}^{n\times n}$ is called a lower semiseparable matrix, if all matrices taken out of the lower triangular part of the matrix A (including the diagonal), have rank at most* 1.

The following property explains the connection between lower semiseparable and Hessenberg matrices, which will play a vital role in the remainder of this manuscript.

Proposition 2 *The inverse of a lower semiseparable matrix A is a Hessenberg matrix H.*

The proof of this proposition can, for example, be found in [11,12].

Moreover, as the inverse of an upper triangular matrix, is again an upper triangular matrix, one might want to replace the role of the Hessenberg matrix in the generalized eigenvalue problem by a lower semiseparable matrix.

In this article if we use a capital letter to denote a matrix (e.g. A) then the corresponding lower case letter with subscript ij refers to the (ij) entry of that matrix (e.g. a_{ij}). We will use a Matlab-like notation to denote submatrices, for example $M(i:j,k:l)$ is the submatrix comprising of the rows $i\dots j$ and columns $k\dots l$ taken out of the matrix M.

In solving the generalized eigenvalue problem $A-\lambda B$ we will need to perform several Givens transformations on the matrices. The Givens transformations that are applied to the left of the matrices, will be denoted by Q_{ij} and Givens transformations, that are applied to the right, by Z_{kl}. Here the subscript indicates which rows or columns are involved in the transformation. When applying consecutive Givens transformations to a matrix M, we add a subscript and a superscript to the notation of the matrix in the following way $M_{ij}^{(k)}$. This means that we have performed i Givens transformations to the left and j Givens transformations to the right of the matrix M in the kth step of the algorithm. Note that we will use M and $M^{(0)}$ or $M_{00}^{(k)}$ and $M^{(k)}$ interchangeably.

Furthermore this manuscript includes several figures of matrices. In these figures we will denote elements of the matrix by $\times$, elements that satisfy the semiseparable structure by $\boxtimes$ and elements that will be annihilated by the next transformation by

$\otimes$.

2 The reduction to lower semiseparable-triangular form

The reduction of a pair of matrices A and B, to Hessenberg-triangular form, via unitary transformations Q and Z is well-known, and can be found for example in [9].

We will now provide a constructive proof for the following theorem.

Theorem 3 *Suppose two $n \times n$ matrices A and B are given. Then there exist, two unitary matrices Q and Z such that $Q^T AZ$ is a lower semiseparable matrix, and $Q^T BZ$ is an upper triangular matrix.*

PROOF. The proof is constructive. An algorithm will be proposed for reducing the pair A and B to lower semiseparable, upper triangular form. We will consider the $n = 6$ case, as it illustrates the general case. We will start by reducing the matrix B to upper triangular form using unitary transformations. These transformations are also applied to the matrix A, in order to preserve the eigenvalues. The resulting matrices are depicted below, where $A^{(0)} = A$, $B^{(0)} = B$ and U^T denotes the transformation responsible for the triangularization of B.

$$A^{(1)}_{00} = U^T A^{(0)}_{00} \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \otimes & \times & \times & \times & \times & \times \end{pmatrix} \qquad B^{(1)}_{00} = U^T B^{(0)}_{00} \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

Since $B^{(1)}$ already has the desired upper triangular form, it suffices to reduce the matrix $A^{(1)}$ to lower semiseparable form, while preserving the form of $B^{(1)}$. First we determine a Givens transformation to annihilate a_{61}. In multiplying both matrices to the left by this Givens transformation, we destroy the upper triangular form of the matrix $B^{(1)}$, as shown below.

$$A^{(1)}_{10} = Q^{(1)T}_{56} A^{(1)}_{00} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \end{pmatrix}, \quad B^{(1)}_{10} = Q^{(1)T}_{56} B^{(1)}_{00} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \otimes & \times \end{pmatrix}$$

We can annihilate the nonzero element in the lower triangular part of $B^{(1)}$, by applying a suitable Givens transformation to the right of both matrices.

$$A_{11}^{(1)} = A_{10}^{(1)} Z_{56}^{(1)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \otimes & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \end{pmatrix}, \quad B_{11}^{(1)} = B_{10}^{(1)} Z_{56}^{(1)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

In a similar manner, we can create additional zeros in the first column of $A^{(1)}$, while preserving the upper triangular structure of the matrix $B^{(1)}$, such that eventually we get the following result:

$$A_{54}^{(1)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \end{pmatrix}, \quad B_{54}^{(1)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ \otimes & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

The final Givens transformation of this step, $Z_{12}^{(1)}$ annihilates b_{21} and simultaneously creates the first non trivial rank one block in the matrix $A^{(2)}$.

$$A_{00}^{(2)} = A_{54}^{(1)} Z_{12}^{(1)} = \begin{pmatrix} \boxtimes & \times & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times & \times \end{pmatrix}, \quad B_{00}^{(2)} = B_{54}^{(1)} Z_{12}^{(1)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

Now assume by induction, that the first three columns of A already satisfy the lower semiseparable structure and we want to create the next low rank block (i.e. add a column to the lower semiseparable structure).

$$A_{00}^{(3)} = \begin{pmatrix} \boxtimes & \times & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times & \times \end{pmatrix}, \quad B_{00}^{(3)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

As we did before, we create zeros in the matrix $A^{(3)}$, while preserving the upper triangular form of the matrix $B^{(3)}$. Because of the rank structure already present in the matrix $A^{(3)}$, applying the first Givens transformation to the left will annihilate three elements in its the bottom row, resulting in the following matrix pair.

$$A_{10}^{(3)} = Q_{56}^{(3)T} A_{00}^{(3)} = \begin{pmatrix} \boxtimes & \times & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \end{pmatrix}, \quad B_{10}^{(3)} = Q_{56}^{(3)T} B_{00}^{(3)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \otimes & \times \end{pmatrix}$$

As in the previous step we should eliminate the nonzero element in the lower triangular part of B by applying the Givens transformation $Z_{56}^{(3)}$. The reader can verify that we can obtain the following matrix pair by applying $Z_{56}^{(3)}$ followed by the application of three additional Givens transformations $Q_{45}^{(3)T}$, $Z_{45}^{(3)}$ and $Q_{34}^{(3)T}$.

$$A_{32}^{(3)} = \begin{pmatrix} \boxtimes & \times & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \end{pmatrix}, \quad B_{32}^{(3)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & \otimes & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

In eliminating the last nonzero element in the lower triangular part of the matrix $B^{(3)}$ we destroy previously created rank one blocks in the matrix $A^{(3)}$. But at the same time we do fill up part of the block of zeros in $A^{(3)}$ hereby creating part of the desired semiseparable structure.

$$A_{33}^{(3)} = A_{32}^{(3)} Z_{34}^{(3)} = \begin{pmatrix} \boxtimes & \times & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times & \times \\ \otimes & \otimes & \times & \times & \times & \times \\ 0 & 0 & \boxtimes & \boxtimes & \times & \times \\ 0 & 0 & \boxtimes & \boxtimes & \times & \times \\ 0 & 0 & \boxtimes & \boxtimes & \times & \times \end{pmatrix}, \quad B_{33}^{(3)} = B_{32}^{(3)} Z_{34}^{(3)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

We know that, in continuing to eliminate elements in the first column(s) of $A^{(3)}$, we will destroy the upper triangular form of $B^{(3)}$. The Givens transformations that restore the upper triangular form of $B^{(3)}$ also restore the semiseparable structure of the lower triangular part of $A^{(3)}$. The following figures illustrate the restoration of part of the semiseparable structure.

$$A_{43}^{(3)} = Q_{23}^{(3)T} A_{33}^{(3)} = \begin{pmatrix} \boxtimes & \times & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & \boxtimes & \boxtimes & \times & \times \\ 0 & 0 & \boxtimes & \boxtimes & \times & \times \\ 0 & 0 & \boxtimes & \boxtimes & \times & \times \end{pmatrix}, \quad B_{43}^{(3)} = Q_{23}^{(3)T} B_{33}^{(3)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & \otimes & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

$$A_{44}^{(3)} = A_{43}^{(3)} Z_{23}^{(3)} = \begin{pmatrix} \boxtimes & \times & \times & \times & \times & \times \\ \otimes & \times & \times & \times & \times & \times \\ 0 & \boxtimes & \boxtimes & \times & \times & \times \\ 0 & \boxtimes & \boxtimes & \boxtimes & \times & \times \\ 0 & \boxtimes & \boxtimes & \boxtimes & \times & \times \\ 0 & \boxtimes & \boxtimes & \boxtimes & \times & \times \end{pmatrix}, \quad B_{44}^{(3)} = B_{43}^{(3)} Z_{23}^{(3)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

Applying suitable Givens transformations $Q_{12}^{(3)T}$ and $Z_{12}^{(3)}$ concludes this step of the reduction and gives us matrices $A^{(4)}$ and $B^{(4)}$ as pictured below.

$$A^{(4)} = \begin{pmatrix} \boxtimes & \times & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times & \times \end{pmatrix}, \quad B^{(4)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

Finally we can create the last non trivial low rank block in a similar way and hereby get the desired reduction. Notice that from now on we will denote the lower semiseparable matrix A by S and the upper triangular matrix B by R.

3 The QZ-algorithm

In this section we will deduce a QZ-method for a pair of matrices S and R, where S is a lower semiseparable matrix and R is an upper triangular matrix. Performing one step of the QZ-method corresponds to performing one step of the QR-algorithm on the matrix $M = SR^{-1}$. Since S is a lower semiseparable matrix and R is an upper triangular matrix we know that the matrix M is lower semiseparable. To perform a QR-step on a semiseparable matrix we need $2n-2$ Givens transformations. These Givens transformations are applied in two consecutive steps. The first step consists of performing a QR-step without shift on the matrix M by applying the first $n-1$ Givens transformations. The next Givens transformation will introduce the shift. The remaining Givens transformations restore the lower semiseparable form of the matrix. For more information concerning the implicit QR-algorithm for semiseparable matrices we refer the reader to [4].

There are two main goals that have to be satisfied by our QZ-algorithm. The first one consists of performing a step of the QR-algorithm on SR^{-1}, without explicitly forming this matrix product. The second one requires that performing a step of the algorithm preserves the structure of the matrix pair (S, R).

Normally, when describing the traditional QZ-iteration one assumes that the Hessenberg matrix is unreduced. If this isn't the case, deflation can be performed. We will need a similar notion of unreducibility for a lower semiseparable matrix. To this end we introduce generator representable lower semiseparable matrices, which are defined as follows.

Definition 4 *A matrix $S \in \mathbb{R}^{n\times n}$ is called a generator representable lower semiseparable matrix, if there exist column vectors u and v such that the following relation is satisfied:*

$$\mathrm{tril}\,(S) = \mathrm{tril}\,(uv^T).$$

In other words, we can say that a lower semiseparable matrix is generator representable, if its lower triangular part comes from a rank 1 matrix.

In this and in the following subsections, we will assume that S is a generator representable lower semiseparable matrix. If this isn't the case the matrix can be divided into several different blocks that are generator representable [11, Proposition 3, p. 845]. In the traditional approach one uses a Hessenberg matrix H instead of a lower semiseparable matrix. The matrices are split up if a subdiagonal element of H equals zero, because in that case an entire subdiagonal block of H is zero. In our approach, we should consider the norm of the whole subdiagonal block to determine whether or not the problem can be divided into two subproblems. More information about the criterion used to determine whether or not the matrices can be divided as well as a fast method to compute the norms, can be found in [4]. Below we will illustrate the easiest case, in which the matrix S has a zero block of dimension $(n-k)\times k$ in the lower left position. In this case $S-\kappa R$, where κ represents the shift, can be divided as follows:

$$S-\kappa R = \begin{bmatrix} \hat{S}_{11}-\kappa\hat{R}_{11} & \hat{S}_{12}-\kappa\hat{R}_{12} \\ 0 & \hat{S}_{22}-\kappa\hat{R}_{22} \end{bmatrix}$$

It should be clear that because of the size of the zero block, $\hat{S}_{11}-\kappa\hat{R}_{11}$ is a $k\times k$ block and $\hat{S}_{22}-\kappa\hat{R}_{22}$ a $(n-k)\times(n-k)$ block. Now it suffices to solve the two smaller problems $\hat{S}_{11}-\kappa\hat{R}_{11}$ and $\hat{S}_{22}-\kappa\hat{R}_{22}$.

Furthermore we will assume that R is nonsingular. If not the matrices can be split up again. We will illustrate this process by means of an example. Consider the following situation, where S and R are both 6×6 matrices and the element r_{33}

equals zero.

$$S=\begin{pmatrix} \boxtimes & \times & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes & \boxtimes \end{pmatrix}, \quad R=\begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

Similar to the reduction in the previous subsection, we will start by creating zeros in the matrix S, from bottom to top, all the while preserving the upper triangular form of R. We will need a total of six Givens transformations to create the situation depicted below. Three of which will be responsible for the creation of the zeros in the structured lower triangular part of S, while the remaining three see to it that R remains upper triangular.

$$S^{(1)}=\left(\begin{array}{ccc|ccc} \boxtimes & \times & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times & \times \\ \hline 0 & 0 & 0 & \boxtimes & \times & \times \\ 0 & 0 & 0 & \boxtimes & \boxtimes & \times \\ 0 & 0 & 0 & \boxtimes & \boxtimes & \boxtimes \end{array}\right), \quad R^{(1)}=\left(\begin{array}{ccc|ccc} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ \hline 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times \end{array}\right)$$

We recognize the above mentioned example, where we could divide the matrices due to a block of zeros in the lower left part of S. Lets call

$$\begin{aligned} \tilde{S}_{11} &= S(1:3,1:3) \\ \tilde{S}_{22} &= S(4:6,4:6) \\ \tilde{R}_{11} &= R(1:3,1:3) \\ \tilde{R}_{22} &= R(4:6,4:6). \end{aligned}$$

The pair $(\tilde{S}_{22},\tilde{R}_{22})$ is already in the desired form. However $\tilde{R}_{11}$ is still singular. We continue to create zeros in S, while preserving the form of R, but this time we only apply the Givens transformation to $\tilde{S}_{11}$ and $\tilde{R}_{11}$. Consider this initial situation:

$$\tilde{S}_{11}=\begin{pmatrix} \boxtimes & \times & \times \\ \boxtimes & \boxtimes & \times \\ \otimes & \otimes & \boxtimes \end{pmatrix}, \quad \tilde{R}_{11}=\begin{pmatrix} \times & \times & \times \\ 0 & \times & \times \\ 0 & 0 & 0 \end{pmatrix}.$$

We apply a Givens transformation to the second and the third row of $\tilde{S}_{11}$ and $\tilde{R}_{11}$. As a result the bottom right 2×2 block of $\tilde{R}_{11}$ fills up, but stays of rank one. Therefore applying a Givens transformation designed to annihilate the element in position

$(3,2)$ of $\tilde{R}_{11}$, will also result in the annihilation of the element in position $(2,2)$.

$$\tilde{S}_{11}^{(1)} = \tilde{Q}_{23}\tilde{S}_{11} = \begin{pmatrix} \boxtimes & \times & \times \\ \boxtimes & \boxtimes & \times \\ 0 & 0 & \boxtimes \end{pmatrix}, \quad \tilde{R}_{11}^{(1)} = \tilde{Q}_{23}\tilde{R}_{11} = \begin{pmatrix} \times & \times & \times \\ 0 & \otimes & \times \\ 0 & \otimes & \times \end{pmatrix}$$

It is easy to verify that two additional Givens transformations suffice to annihilate the elements in the first column of the lower triangular part of $\tilde{S}_{11}$, while preserving the upper triangular form of $\tilde{R}_{11}$. This results in the following matrix pair:

$$\tilde{S}_{11}^{(2)} = \tilde{S}_{11}^{(1)}\tilde{Z}_{23} = \left(\begin{array}{c|cc} \times & \times & \times \\ \hline 0 & \boxtimes & \times \\ 0 & \boxtimes & \boxtimes \end{array}\right), \quad \tilde{R}_{11}^{(2)} = \tilde{R}_{11}^{(1)}\tilde{Z}_{23} = \left(\begin{array}{c|cc} 0 & \times & \times \\ \hline 0 & \times & \times \\ 0 & 0 & \times \end{array}\right).$$

The first column of $\tilde{R}_{11}^{(2)}$ consists entirely of zeros, while in the corresponding column of $\tilde{S}_{11}^{(2)}$ only the diagonal element differs from zero. Therefore we have found an infinite generalized eigenvalue. Now we can again deflate and continue with $\tilde{S}_{11}^{(2)}(2:3,2:3)$ and $\tilde{R}_{11}^{(2)}(2:3,2:3)$. The case where S is singular can be solved in a similar manner.

3.2 Part I

In this subsection we will describe the first part of the QZ-algorithm, in which a QR-step without shift is performed on the matrix $M = SR^{-1}$. In order to avoid the construction of the matrix M, we will apply the Givens transformations to S and R separately. We need only $n-1$ Givens transformations to reduce S to upper triangular form, because of its semiseparable structure. As in the previous sections these transformations are applied from bottom to top to both S and R. But this time, no extra Givens transformations are applied in order to preserve the triangular form of R. Hence at the end of this first step we have transformed our original semiseparable-triangular matrix pair (S,R) into a triangular-Hessenberg pair $(S^{(1)},R^{(1)})$.

3.3 Part II

Next we have to introduce a shift in order to speed up the convergence of the implicit QR- algorithm. We could for example use the Rayleigh shift or the Wilkinson shift. The actual choice of the shift isn't relevant for the understanding of the algorithm. Therefore we leave details about the choice of the shift to the section about numerical experiments and in the remainder of this section we shall refer to the shift by κ.

Let us denote the Givens transformations of the previous subsection by $G_1, G_2 \dots G_{n-1}$. If we were to explicitly compute $G_{n-1} \dots G_1(SR^{-1} - \kappa I)$ we would get an upper Hessenberg matrix H. The next Givens transformation G_n is responsible for the annihilation of the element h_{21}. Let $S^{(1)} = G_{n-1} \dots G_1 S$. Then we can calculate the first column of H as the sum of

$$\begin{aligned}
S^{(1)}R^{-1}e_1 &= S^{(1)}\frac{1}{b_1} && \text{and} \\
G_{n-1}\dots G_1 \kappa I e_1 &= G_{n-1}\kappa I e_1 \\
&= \begin{bmatrix} c_{n-1}\kappa & -\bar{s}_{n-1}\kappa & 0 \dots 0 \end{bmatrix}^T, && \text{with} \\
G_{n-1} &= \begin{bmatrix} c_{n-1} & s_{n-1} \\ -\bar{s}_{n-1} & c_{n-1} \end{bmatrix}.
\end{aligned}$$

Now we can easily determine the Givens transformation G_n responsible for the annihilation of the second element of this columnvector. We will now again refer to the Givens transformations using Q_{ij} and Z_{kl}. The following figure illustrates the matrix pair after applying the nth Givens transformation to the first two rows.

$$S_{10}^{(1)} = Q_{12}^{(1)T} S_{00}^{(1)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times \end{pmatrix}, \quad R_{10}^{(1)} = Q_{12}^{(1)T} R_{00}^{(1)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \end{pmatrix}$$

We have created a bulge in the lower triangular part of $S^{(1)}$, which will be annihilated by the next Givens transformation as illustrated below.

$$S_{11}^{(1)} = S_{10}^{(1)} Z_{12}^{(1)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ \otimes & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times \end{pmatrix}, \quad R_{11}^{(1)} = R_{10}^{(1)} Z_{12}^{(1)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \end{pmatrix}$$

Note that while $Z_{12}^{(1)}$ annihilated the bulge in $S^{(1)}$, it created a new bulge in $R^{(1)}$, which will have to be annihilated next.

$$S_{21}^{(1)} = Q_{23}^{(1)T} S_{11}^{(1)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times \end{pmatrix}, \quad R_{21}^{(1)} = Q_{23}^{(1)T} R_{11}^{(1)} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \otimes & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \end{pmatrix}$$

By multiplying both matrices with $Q_{23}^{(1)T}$ a new bulge is created in $S^{(1)}$ and we're back at our starting position, only now the bulge is situated lower than the original bulge. The next series of Givens transformations will chase the bulges in $S^{(1)}$ and $R^{(1)}$ down so that in the end we get the triangular-Hessenberg matrix pair $(S^{(2)}, R^{(2)})$. Lastly we have to reduce $R^{(2)}$ to upper triangular form. We create the necessary zeros by applying $n-1$ Givens transformations to the right of the matrices, from back to front. In doing so, we also create the semiseparable structure of the matrix $S^{(2)}$.

This gives us a semiseparable-triangular matrix pair and ends the QZ-step.

3.4 Proof of correctness of QZ algorithm

We will prove that in the previous subsections we in fact performed a step of the QR-algorithm. Performing the traditional QR-method on $(SR^{-1} - \kappa I)$ gives us a lower semiseparable matrix C_1, with

$$C_1 = Q_1^T (SR^{-1}) Q_1.$$

Our approach results in two new matrices S_2 and R_2 defined by:

$$\begin{aligned} S_2 &= Q_2^T S Z_2 \\ R_2 &= Q_2^T R Z_2, \end{aligned}$$

with Q_2 and Z_2 the product of all the Givens transformations that have been performed to the left and right respectively. We know that (S_2, R_2) is a (lower semiseparable, upper triangular) matrix pair. Therefore if we calculate $S_2 R_2^{-1}$, the resulting matrix C_2 is lower semiseparable. Furthermore we have:

$$\begin{aligned} C_2 &= Q_2^T S Z_2 Z_2^{-1} R^{-1} Q_2, \\ &= Q_2^T S R^{-1} Q_2 \\ C_1 &= Q_1^T (SR^{-1}) Q_1. \end{aligned}$$

We also know that $Q_1 e_1 = Q_2 e_1$ due to construction.

This implies, due to the implicit Q-theorem for semiseparable matrices [13], that we indeed performed a step of QR on the pair (S,R).

3.5 A second look at the QZ-algorithm

The QZ-algorithm as developed in this section is based on the QR-algorithm for the corresponding SR^{-1} problem. As told before the QR-factorization of the matrix SR^{-1} consists of performing two sequences of Givens transformations, one upgoing and one descending. After having performed the upgoing sequence the remaining Givens transformations are determined in an implicit way.

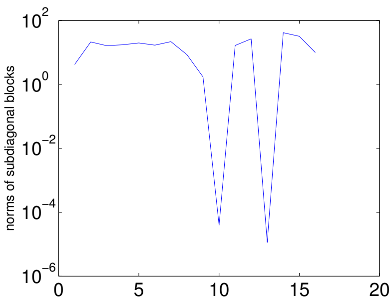
Taking a closer look at the QZ-algorithm above, we notice that performing the first sequence of transformations from bottom to top transforms our (S,R) pair into a pair (U,H), with U and H respectively upper triangular and Hessenberg. Following this first sequence of Givens transformations, in fact a kind of QZ-step for the traditional case is performed on the matrices (U,H). Finally the last sequence of Givens transformations transforms the resulting matrices back to the semiseparable-triangular form.

In fact one can change from the (S,R) pair to a pair (U,H) and compute the eigenvalues of this pair via the traditional QZ-algorithm. This means that we can combine the advantages of the reduction to semiseparable form and the speed and knowledge of the traditional QZ-algorithm. One has to be careful however because w.r.t. the traditional QZ-algorithm applied on a Hessenberg-triangular pair we work here on a triangular-Hessenberg pair, which causes in some sense an inverse convergence behavior.

To conclude we might say that the QZ-algorithm for semiseparable-triangular matrices also uses as an intermediate step the QZ-algorithm for Hessenberg-triangular (or triangular-Hessenberg) matrices. We remark that this is not the case in the traditional QR-algorithm for semiseparable matrices, w.r.t. the QR-algorithm for tridiagonal matrices. This insight opens the possibility for combining the advantages of the reduction method with the speed and knowledge of the standard QZ-algorithm. This is work in progress.

4 Numerical experiments

In this section two numerical experiments are performed. The first experiment illustrates the convergence properties of the reduction algorithm. In the second experiment, the accuracy of the eigenvalues found with our QZ-algorithm are compared to those found by Matlab.



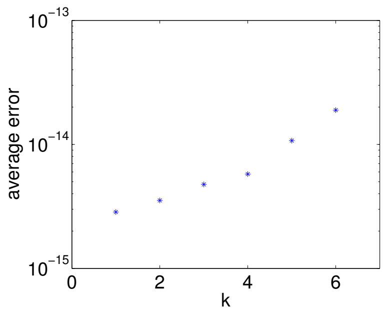


Fig. 1. The figure on the left contains the norms of the subdiagonal blocks of the matrix S. The average errors of the eigenvalues for matrices of dimension 10.2^k are pictured in the figure on the right.

For the first experiment we constructed a matrix pair (A,B) such that the generalized eigenvalues equal

$$1,\ldots,10,100,101,102,1000,1001,1002,1003.$$

After reduction, we get the semiseparable-triangular matrix pair (S,R). In the left graph of Figure 1 we plotted the norms of the subdiagonal blocks of S. The norms of $S(11:17,1:10)$ and $S(14:17,1:13)$ are negligible. This implies that $S(11:13,11:13)$ and $S(14:17,14:17)$ can already be deflated before performing the first step of the QZ.

A set of test matrices was generated of dimensions 10.2^k for $i=0,\ldots,6$ for the second experiment. Since we wanted to be able to compute complex eigenvalues, the Wilkinson shift was used in the implementation of the QZ-algorithm. On average the algorithm needed between 3 and 3.6 iterations to find an eigenvalue. The average relative error of the eigenvalues found by our algorithm compared to those found by Matlab for the different dimensions is pictured in Figure 1 on the right.

5 Conclusions

In this manuscript we proposed a new method for computing the generalized eigenvalues of a pair of matrices. The newly described method makes use of structured rank matrices instead of sparse matrices. The algorithm consists naturally of two steps. In a first step a reduction to structured rank form is presented. Important in this reduction is a kind of inherited nested subspace iteration, which is performed on the matrices. The second step consists of applying the QZ-algorithm to the involved structured rank matrices. We remark that the rank structured QZ-algorithm seems to be closely related to a kind of “inverted” standard QZ-algorithm. This is work in progress.

References

[1] 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.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

[6] M. Van Barel, R. Vandebril, and N. Mastronardi. An orthogonal similarity reduction of a matrix into semiseparable form. *SIAM Journal on Matrix Analysis and its Applications*, 27(1):176–197, 2005.

[7] R. Vandebril, E. Van Camp, M. Van Barel, and N. Mastronardi. On the convergence properties of the orthogonal similarity transformations to tridiagonal and semiseparable (plus diagonal) form. *Numerische Mathematik*, 104(TW436):205–239, 2006.

[8] G. W. Stewart. *Matrix Algorithms, Vol II Eigensystems*. SIAM, 1999.

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

[10] F. Tisseur and K. Meerbergen. The quadratic eigenvalue problem. *SIAM Review*, 43(2):235–286, June 2001.

[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] M. Fiedler. Structure ranks of matrices. *Linear Algebra and its Applications*, 179:119–127, 1993.

[13] R. Vandebril, M. Van Barel, and N. Mastronardi. An implicit Q theorem for Hessenberg-like matrices. *Mediterranean Journal of Mathematics*, 2:59–275, 2005.