Electronic Transactions on Numerical Analysis.
Volume 33, pp. 126-150, 2009.

ISSN 1068-9613.

ETNA
Kent State University
http://etna.math.kent.edu

A NEW ITERATION FOR COMPUTING THE EIGENVALUES OF SEMISEPARABLE (PLUS DIAGONAL) MATRICES*

RAF VANDEBRIL†, MARC VAN BAREL†, AND NICOLA MASTRONARDI‡

Abstract. This paper proposes a new type of iteration for computing eigenvalues of semiseparable (plus diagonal) matrices based on a structured-rank factorization. Remarks on higher order semiseparability ranks are also made. More precisely, instead of the traditional QR iteration, a QH iteration is used. The QH factorization is characterized by a unitary matrix Q and a Hessenberg-like matrix H in which the lower triangular part is semiseparable (often called a lower semiseparable matrix). The Q factor of this factorization determines the similarity transformation of the QH method.

It is shown that this iteration is extremely useful for computing the eigenvalues of structured-rank matrices. Whereas the traditional QR method applied to semiseparable (plus diagonal) and Hessenberg-like matrices uses similarity transformations involving $2p(n-1)$ Givens transformations (where p denotes the semiseparability rank), the QH iteration only needs $p(n-1)$ Givens transformations, which is comparable to the generalized Hessenberg (symmetric band) situation having p subdiagonals. It is also shown that this method can in some sense be interpreted as an extension of the traditional QR method for Hessenberg matrices, i.e., the traditional case also fits into this framework. It is also shown that this iteration exhibits an extra type of convergence behavior compared to the traditional QR method.

The algorithm is implemented in an implicit way, based on the Givens-weight representation of the structured rank matrices. Numerical experiments show the viability of this approach. The new approach yields better complexity and more accurate results than the traditional QR method.

Key words. QH algorithm, structured rank matrices, implicit computations, eigenvalue, QR algorithm, rational QR iteration

AMS subject classifications. 65F05

1. Introduction and preliminary results.

Many authors are currently investigating efficient algorithms for computing the eigenvalues of structured rank matrices. All the methods discussed thus far focus attention on QR algorithms for computing the eigenvalues of these matrices. Various QR-type algorithms exist for higher order structured rank matrices, generalized eigenvalue problems, polynomial root finding algorithms and so forth [2, 4–7, 11, 20].

The QR factorization of a Hessenberg (tridiagonal[1]) matrix can be computed easily by performing a sequence of $n-1$ Givens transformations from top to bottom, annihilating in each of the $n-1$ steps one subdiagonal element [13, 14]. The corresponding (single shift) implicit QR algorithm also uses $n-1$ Givens transformations. The implicit version consists of an initial Givens similarity transformation applied to the Hessenberg (tridiagonal) matrix. This introduces a disturbing element, the so-called bulge, in the structure. In the implicit version, one constructs the remaining $n-2$ Givens transformations so that the bulge

* Received December 31, 2007. Accepted March 31, 2009. Published online on December 9, 2009. Recommended by A. Salam. The research of the first two authors, was partially supported by the Research Council K.U. Leuven, project OT/05/40 (Large rank structured matrix computations), CoE EF/05/006 Optimization in Engineering (OPTEC), by the Fund for Scientific Research–Flanders (Belgium), 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 Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office, Belgian Network DYSCO (Dynamical Systems, Control, and Optimization). The first author has a grant as "Postdoctoraal Onderzoeker" from the Fund for Scientific Research–Flanders (Belgium). The work of the third author was partially supported by MIUR, grant number 2004015437. The scientific responsibility rests with the authors.

†K.U.Leuven, Dept. Computerwetenschappen, ({raf.vandebril,marc.vanbarel}@cs.kuleuven.be)

‡Istituto per le Applicazioni del Calcolo M. Picone, sez. Bari, (n.mastronardi@ba.iac.cnr.it)

[1] When discussing tridiagonal and semiseparable matrices in the context of eigenvalue computations, we assume them to be symmetric.

ETNA
Kent State University
http://etna.math.kent.edu

is removed and we obtain again a Hessenberg (tridiagonal) matrix [33]. Implicitly, one has now performed a step of the shifted QR method.

The QR factorization of a semiseparable (Hessenberg-like) matrix plus a diagonal[2] consists of $2n-2$ Givens transformations [17]. A first sequence of Givens transformations from bottom to top transforms the semiseparable (Hessenberg-like) plus diagonal matrix into a Hessenberg matrix, whereas the second sequence of transformations from top to bottom brings the Hessenberg matrix to upper triangular form. The implicit QR algorithm connected to this type of QR factorization also can be decomposed into two steps. A first step corresponds to a similarity transformation involving $n-1$ Givens transformations; see [11, 20]. In the second step, a disturbance is introduced and $n-2$ Givens transformations are needed to restore the structure. Unfortunately, this implicit QR algorithm uses twice as many Givens transformations as the corresponding algorithm for the Hessenberg (tridiagonal) case.

This paper introduces a new type of algorithm for computing the eigenvalues of structured rank matrices. The new algorithm is based on a so-called QH factorization. This is a factorization of a matrix $A = \check{Q}\check{Z}$, in which $\check{Q}$ is unitary and $\check{Z}$ is a Hessenberg-like matrix (in which the lower triangular part of the matrix has semiseparable form). This unitary matrix $\check{Q}$ is used to define the new iterate $A_{QH} = \check{Q}^H A \check{Q}$. It is shown that this iteration can be performed in an efficient manner for structured rank matrices. More precisely, the QH factorization of a Hessenberg-like minus shift matrix $Z - \mu I$ also consists of $n-1$ Givens transformations. The QH algorithm also can be implemented in an implicit way, such that $n-1$ Givens transformations instead of the traditional $2n-2$ are needed. Besides the fact that the method is cheaper in terms of numerical computations for structured rank matrices, we also show that this new iteration inherits a new type of convergence behavior, which can be advantageous in many cases.

The paper is organized as follows. This section continues by briefly introducing the classes of semiseparable, Hessenberg-like (plus diagonal) matrices as well as the Givens-weight representation. In Section 2, various methods for computing the QR factorization of structured rank matrices are introduced. Based on these different types of QR factorizations, one can deduce different types of QR algorithms. The different ways of computing these QR algorithms are discussed in Section 2.4. Section 3 discusses the QH factorization, which is the basis for the new QH method. A rigorous treatment of the convergence and preservation of structure is presented in Section 4. An implicit version of the method for Hessenberg-like plus diagonal matrices is presented in Section 5. Before providing numerical experiments in Section 7, we briefly show that the QR method for Hessenberg matrices can be considered as a special case of the QH method. This is done in Section 6.

1.1. Definitions. The class of semiseparable and Hessenberg-like matrices considered in this paper is defined as follows.

DEFINITION 1.1. *A square matrix S is called a $\{p,q\}$-semiseparable matrix if the following relations are satisfied:*

$$\operatorname{rank} S(1:i+q-1, i:n) \leq q \quad \textit{and} \quad \operatorname{rank} S(i:n, 1:i+p-1) \leq p,$$

for all feasible i. A matrix is called $\{p\}$-semiseparable if it is $\{p,p\}$-semiseparable, and semiseparable if it is $\{1,1\}$-semiseparable.

DEFINITION 1.2. *A square matrix Z is called a $\{p\}$-Hessenberg-like (or lower semiseparable) matrix if the following relations are satisfied:*

$$\operatorname{rank} Z(i:n, 1:i+p-1) \leq p,$$

[2]The diagonal is necessary for introducing the shift matrix $-\mu I$ in the shifted QR algorithm. In the Hessenberg (tridiagonal) case this does not influence the structure, whereas in the structured rank case it does.

for all feasible i.

Sometimes $\{p\}$-generalized Hessenberg matrices arise. These matrices are extensions of the standard Hessenberg matrices, and have $\{p\}$ subdiagonals different from zero.

For simplicity, we focus on Hessenberg-like (plus diagonal) matrices in this paper. There is no loss of generality, because only the structure of the lower triangular part of the involved matrices is important in the theoretical analysis. Hence, for most derivations, we do not need to know the structure of the upper triangular part. This is very important for actual implementations in order to obtain the lowest possible computational complexity. The QR algorithm computes QR factorizations of the matrices $Z - \mu I$ for the shifted Hessenberg-like matrix, or $Z + D - \mu I$ for the shifted Hessenberg-like plus diagonal matrix. Since both shifted matrices are essentially Hessenberg-like plus diagonal matrices, we discuss in the next section the QR factorization of a Hessenberg-like plus diagonal matrix.

1.2. Representation. The matrices defined above are dense in the sense that they contain mostly nonzero elements. But these matrices can be represented by using only a limited number of parameters. They admit, for example, a sparse representation based on Givens transformations. This representation is the so-called Givens-weight representation for the general structured rank case (see [8]), or the Givens-vector representation for the class of $\{1\}$-semiseparable matrices[3]. More precisely, the Givens-weight representation for the lower triangular part of a $\{p\}$-Hessenberg-like matrix Z consists of p sequences of Givens transformations. In fact, it is a sort of QR factorization of the matrix:

$$Q_1^H Q_2^H \dots Q_p^H Z = R \text{ and } Z = Q_1 Q_2 \dots Q_p R = QR, \tag{1.1}$$

where every unitary matrix Q_i^H consists of $(n-1)-(p-i)$ Givens transformations, peeling off a rank-1 part from the Hessenberg-like matrix Z. Each of the matrices Q_i contains a descending sequence of Givens transformations. This means that for a particular Q_i, the first Givens transformation acts on rows $p-i+1$ and $p-i+2$, the second on rows $p-i+2$ and $p-i+3$, and so forth. They start changing the top rows of the matrix and go downwards; hence, the name descending. Similarly, we call the sequence corresponding to Q_i^H ascending.

In an actual implementation, one does not really store the matrix R, but a condensed form (called the weights). The effective representation consists of p sequences of Givens transformations plus the weights.

One can also construct such a representation for the upper triangular part, if it has rank structure. In the case of a $\{p, q\}$-semiseparable matrix, one has p sequences of Givens transformations for storing the lower triangular part and q sequences for storing the upper triangular part plus all weights. The use of the weights is only necessary for implementation details. For theoretical purposes, we work with the QR-like formulation from (1.1). More information can be found in [8, 21].

The above representation is often referred to as the top-bottom representation, as it starts on the top row of the matrix R (right equation in (1.1)) and gradually fills up the matrix from the top to the bottom. One can easily change this representation to another kind of factorization: $Z = RQ$, where the matrix Q consists again of p sequences of Givens transformations, now gradually filling up the low rank part of the matrix from right to left. This is called a right-left representation. One can easily convert from the top-bottom form to the right-left form in $O(pn)$ flops[4].

[3]There are many more representations, such as the quasiseparable, generator representation and so forth.
[4]Every operation of the form $+, -, /, *$ is considered as a flop.

ETNA
Kent State University
http://etna.math.kent.edu

2. The QR factorization and its variants. The idea for the new iteration finds its origin in the different variants for computing the QR factorization of structured rank matrices. These variants result, of course, in different QR algorithms. Let us briefly discuss the different forms for computing the QR factorization of structured rank matrices. For simplicity we assume we are working with a Hessenberg-like plus diagonal matrix; semiseparable plus diagonal matrices and higher order semiseparable plus diagonal matrices can be treated in the same way.

2.1. The traditional factorization: $\wedge$ pattern. For this type of QR factorization, an ascending sequence of Givens transformations is applied to the Hessenberg-like plus diagonal matrix $Z+D$, followed by a descending sequence of Givens transformations. More information on this type of QR factorization can be found in [9, 10, 17, 22]. The first ascending sequence of Givens transformations acting on $Z+D$, denoted by $\hat{Q}_1^H$ consists of $n-1$ Givens transformations in which each Givens transformation acts on two successive rows of the matrix Z, exploiting thereby the rank structure in the lower triangular part to annihilate all elements below the diagonal (these unitary transformations coincide with the ones from the top to bottom representation). We obtain

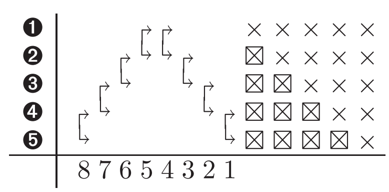
$$\hat{Q}_1^H Z = R \quad \text{and} \quad \hat{Q}_1^H (Z+D) = H,$$

in which H is a Hessenberg matrix. This is followed by a second sequence of $n-1$ Givens transformations from top to bottom to annihilate the subdiagonal elements of the matrix H. This gives

$$\hat{Q}_2^H H = \hat{Q}_2^H \hat{Q}_1^H (Z+D) = \hat{Q}^H (Z+D) = \hat{R}, \tag{2.1}$$

in which $\hat{R}$ is the resulting upper triangular matrix. This is the standard QR factorization, which is discussed in detail in the paper [17].

We often work with a graphical interpretation related to Givens transformations and the matrix they are acting on. The matrix product $\hat{Q}^H(Z+D)$ is graphically represented as follows.



(2.2)

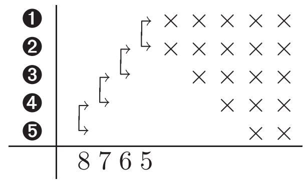
The right part consisting of $\times$ and $\boxtimes$ elements represents the matrix $Z+D$. The elements $\boxtimes$ denote the part of the matrix satisfying the rank structure. The elements $\times$ denote arbitrary elements. In this figure, the elements on the diagonal cannot be included in the rank structure because they are perturbed by the diagonal D. The left part, consisting of the brackets with arrows, denotes the Givens transformations.

The numbered circles on the vertical axis depict the rows of the matrix, to indicate on which rows the Givens transformations act. The bottom numbers represent in some sense a time line to indicate in which order the Givens transformations are performed. The brackets in the table represent graphically a Givens transformation acting on the rows in which the arrows of the brackets are lying. The Givens transformations from columns 1 up to 4 represent the Givens transformations in the matrix $\hat{Q}_1^H$. The ones in the columns 5 up to 8 denote these of the matrix $\hat{Q}_2^H$; see (2.1).

Let us explain this schemes in more detail. First, a Givens transformation is performed, the one in position 1 in Scheme 2.2, that acts on row 5 and row 4 to annihilate the first three

ETNA
Kent State University
http://etna.math.kent.edu

elements of row 5. Second, a Givens transformation is performed that acts on row 3 and row 4 to annihilate the first two elements of row 4, and this process continues. Applying the Givens transformations in positions 1 through 4 to the matrix on the right results in the following graphical representation. This represents exactly the same matrix as in the previous scheme, but equals now $\hat{Q}_2^H H$.



(2.3)

Applying the remaining four Givens transformations in Scheme 2.3 to the Hessenberg matrix on the right removes the remaining subdiagonal elements. Hence, we obtain the upper triangular matrix $\hat{R}$. Therefore, Scheme 2.2 gives a graphical way to represent the QR factorization of a Hessenberg-like plus diagonal matrix.

NOTE 2.1. *Consider a $\{p\}$-Hessenberg-like plus diagonal matrix. First, one removes the low rank part by applying p ascending sequences of Givens transformations. This gives us*

$$\hat{Q}_p^H \ldots \hat{Q}_1^H (Z + D) = R + H,$$

in which H is a generalized Hessenberg matrix, having p nonzero subdiagonals. To complete the QR factorization, another p top-to-bottom sequences of Givens rotations are needed, each of which removes one subdiagonal from H.

Globally, we have p ascending sequences of Givens transformations for removing the rank p structure, followed by p descending sequences of Givens transformations removing the p subdiagonals. This leads again to a so-called $\wedge$ pattern, this one having thicker legs.

Due to some specific properties of Givens transformations we can obtain other patterns, as we describe in the next two subsections.

2.2. Some properties of Givens transformations. Briefly, two important properties of Givens transformations are mentioned here. We also show their graphical interpretation.

LEMMA 2.2. *Suppose two Givens transformations*[5] *G_1 and G_2 are given:*

$$G_1 = \begin{bmatrix} c_1 & -\bar{s}_1 \\ s_1 & \bar{c}_1 \end{bmatrix} \text{ and } G_2 = \begin{bmatrix} c_2 & -\bar{s}_2 \\ s_2 & \bar{c}_2 \end{bmatrix}.$$

Then we have that $G_1 G_2 = G_3$ is again a Givens transformation. We call this the fusion of Givens transformations in the remainder of the text.

The proof is trivial. In our graphical schemes, we depict this as follows.

The following lemma is very powerful and allows us to interchange the order of Givens transformations and to obtain different patterns. Quite often Givens transformations of higher

[5] The considered transformations are in fact rotations. More information on Givens rotations can be found in [3].

ETNA
Kent State University
http://etna.math.kent.edu

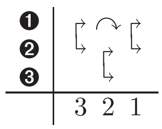
dimensions, say n, are considered. This means that the corresponding 2×2 Givens transformation is embedded in the identity matrix of dimension n, still changing only two consecutive rows when applied to the left.

LEMMA 2.3 (Shift through lemma). *Suppose three 3×3 Givens transformations $\check{G}_1, \check{G}_2$ and $\check{G}_3$ are given, such that the Givens transformations $\check{G}_1$ and $\check{G}_3$ act on the first two rows of a matrix, and $\check{G}_2$ acts on the second and third row (when applied on the left to a matrix). Then there exist Givens transformations $\hat{G}_1, \hat{G}_2$, and $\hat{G}_3$ such that*

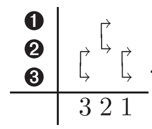
$$\check{G}_1 \check{G}_2 \check{G}_3 = \hat{G}_1 \hat{G}_2 \hat{G}_3,$$

where $\hat{G}_1$ and $\hat{G}_3$ work on the second and third row and $\hat{G}_2$ works on the first two rows.

This result is well-known. The proof can be found in [22] and is simply based on the fact that one can factorize a 3×3 unitary matrix in different ways. Graphically we depict this rearrangement as follows.



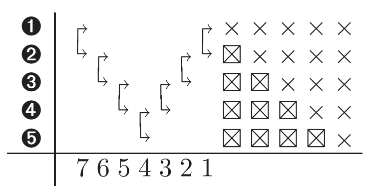
resulting in



Of course, there is a similar transformation that transforms the right figure to the left figure, which we would depict by a $\circlearrowleft$ in the right figure.

2.3. The $\vee$ pattern. We now show how one can change the order of the Givens transformations in Scheme 2.2. We ultimately obtain a different graphical scheme that represents exactly the same factorization, but in which the Givens transformations are performed in a different order.

After applying Lemma 2.2 to the Givens transformations in position 4 and 5 in Scheme 2.2, we can apply the shift through lemma several times (three times in this case), and thereby change the order of the transformations so that we obtain the following factorization.



(2.4)

This gives us the $\vee$ pattern for computing the QR factorization of a matrix. The order of the Givens transformations has changed, but we compute the same QR factorization (more information can be found in [26]):

$$\check{Q}_2^H \check{Q}_1^H (Z + D) = \hat{R}.$$

NOTE 2.4. *Some important remarks related to the $\vee$ and $\wedge$ patterns must be made.*

- *We have the equality*

$$\check{Q}_1 \check{Q}_2 = \hat{Q}_1 \hat{Q}_2;$$

since $\hat{R}$ was not affected, we obtain an identical QR factorization.

- *But generically:*

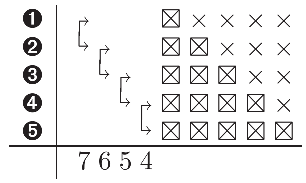
$$\check{Q}_2 \neq \hat{Q}_2$$
$$\check{Q}_1 \neq \hat{Q}_1,$$

ETNA
Kent State University
http://etna.math.kent.edu

which means that the factorization of the unitary matrix in the QR factorization is different in the two patterns.

This pattern can also be decomposed into two parts. First, a descending sequence of Givens transformations (position 1 up to 3) is applied, followed by an ascending sequence of Givens transformations (position 4 up to 7). To distinguish between the $\vee$ and the $\wedge$ pattern we put a $\vee$ on top of the unitary transformations in case of the $\vee$ pattern.

The first three Givens transformations are, in fact, rank expanding Givens transformations. They lift up the rank structure. Hence, after having applied these first Givens transformations, we obtain the following scheme.



(2.5)

The figure clearly illustrates that the strictly lower triangular rank structure has lifted up and that the diagonal may be included in the lower triangular rank structure.

The remaining four Givens transformations from bottom to top remove the rank-1 structure in the lower triangular part so that we obtain the upper triangular matrix $\hat{R}$.

Writing the above figure in mathematical formulas, we obtain

$$\begin{aligned} \check{Q}_2^H \check{Q}_1^H (Z+D) &= \check{Q}_2^H \check{Z}, \\ \check{Q}_1^H (Z+D) &= \check{Z}, \\ (Z+D) &= \check{Q}_1 \check{Z}, \end{aligned}$$

where $\check{Z}$ denotes a Hessenberg-like matrix. The final equation denotes a structured rank factorization of the matrix $Z+D$, since the matrix $\check{Z}$ is of Hessenberg-like form and $\check{Q}_1$ is a unitary transformation. This unitary-Hessenberg-like (QH) factorization forms the basis of the eigenvalue computations proposed in this paper.

DEFINITION 2.5. *A factorization of the form*

$$A = \check{Q}\check{Z},$$

with $\check{Q}$ unitary and $\check{Z}$ a Hessenberg-like matrix is called a unitary-Hessenberg-like factorization, or a QH factorization. In the case that the matrix $\check{Z}$ is a $\{p\}$-Hessenberg-like matrix, we still call this a QH factorization, but we specify the rank of the matrix $\check{Z}$.

NOTE 2.6. *This factorization is a straightforward extension of the QR factorization, as the QR factorization is a QH factorization in which the matrix $\check{Z}$ is of semiseparability rank* 0*, i.e., the strictly lower triangular part of $\check{Z}$ is zero.*

NOTE 2.7. *For a $\{p\}$-Hessenberg-like plus diagonal matrix $Z+D$ we will use a higher order QH factorization in which $\check{Z}$, the Hessenberg-like matrix, has a lower triangular part of $\{p\}$-Hessenberg-like form. More precisely, in this case, one needs $\mathcal{O}(p(n-1))$ Givens transformations for obtaining the factorization. To prove this statement one has to combine Note 2.1 and the results from this subsection.*

2.4. The QR algorithm and its variants. As there are different manners of computing the QR factorization, the QR algorithms are slightly different. In fact, one obtains exactly the same result, but the way of computing the matrices after one step of the QR method can differ. In this section, we will briefly discuss the QR algorithms associated with both the $\wedge$ and the $\vee$ patterns for computing the QR factorization. We remark once more that the

ETNA
Kent State University
http://etna.math.kent.edu

final outcome of both transformations is equal; however, there are differences both in the order in which the Givens transformations are performed and in the Givens transformations themselves.

2.4.1. The QR algorithm connected to the $\wedge$ pattern. We consider the following iteration step on a Hessenberg-like minus shift matrix:

$$\begin{aligned} Z - \mu I &= \hat{Q}_1 \hat{Q}_2 \hat{R}, \\ Z_{QR} &= \hat{R}\hat{Q}_1\hat{Q}_2 + \mu I = \hat{Q}_2^H \hat{Q}_1^H Z \hat{Q}_1 \hat{Q}_2, \end{aligned}$$

in which Z_{QR} denotes the new iterate. We comment on the Hessenberg-like plus diagonal case afterward.

The single shift QR algorithm based on the $\wedge$ pattern was first discussed in an implicit form in [20].

Let us discuss the global flow of the iteration related to the $\wedge$ pattern. The iteration can be decomposed into two steps, each step corresponding to performing a sequence of $n-1$ Givens transformations. The first sequence is an ascending one denoted by $\backslash$ in the $\wedge$ pattern, which annihilates the low rank part in the Hessenberg-like matrix. The second sequence corresponds to the descending Givens transformations denoted by $/$ in the $\wedge$ pattern, which removes the subdiagonal elements.

Since the new iterate is defined as $\hat{Q}_2^H \hat{Q}_1^H Z \hat{Q}_1 \hat{Q}_2 = \hat{Q}_2^H (\hat{Q}_1^H Z \hat{Q}_1) \hat{Q}_2$, two similarity transformations need to be applied to the matrix Z. One is determined by $\hat{Q}_1$ and the other by $\hat{Q}_2$.

- The first similarity transformation (related to $\hat{Q}_1$) computes the following (see Subsection 2.1):

$$\tilde{Z} = \hat{Q}_1^H Z \hat{Q}_1 = \left(\hat{Q}_1^H Z\right) \hat{Q}_1 = R\hat{Q}_1.$$

  This corresponds to performing a step of the QR method without shift on the matrix Z. As a result, we obtain another Hessenberg-like matrix $\tilde{Z}$.
- The second similarity transformation (related to $\hat{Q}_2$) can be performed in an implicit way as follows. Determine the first Givens transformation $\tilde{G}$ of $\hat{Q}_2$ to annihilate the element in position $(2,1)$ of the Hessenberg matrix $\hat{Q}_1^H(Z - \mu I) = H$. Applying this Givens transformation $\tilde{G}$ as a similarity transformation on the Hessenberg-like matrix $\tilde{Z}$ disturbs the specific rank structure of this Hessenberg-like matrix. The implicit part of the method consists of finding the remaining $n-2$ Givens transformations and applying them to $\tilde{G}^H \tilde{Z} \tilde{G}$ so that the resulting matrix is back in Hessenberg-like form. Based on the implicit Q theorem for Hessenberg-like matrices, one knows that this approach results in a Hessenberg-like matrix that is essentially the same as the one resulting from an explicit step of the QR method.

NOTE 2.8. *The first similarity transformation based on $\hat{Q}_1$ is independent of the chosen shift μ. The second similarity transformation is dependent on the shift μ.*

The QR method for Hessenberg-like plus diagonal matrices $Z + D$ is identical. One first performs a number of Givens transformations, corresponding to a step of QR-without shift on Z, followed by a similarity transformation determined by $\hat{Q}_2$. To restore the structure in the Hessenberg-like plus diagonal case, one needs to take into consideration the structure of the diagonal, as the diagonal is preserved under a step of the QR method [16].

2.4.2. The QR algorithm connected to the $\vee$ pattern. We consider the iteration step:

$$\begin{aligned} Z - \mu I &= \check{Q}_1 \check{Q}_2 \hat{R}, \\ Z_{QR} &= \hat{R}\check{Q}_1\check{Q}_2 + \mu I = \check{Q}_2^H \check{Q}_1^H Z \check{Q}_1 \check{Q}_2, \end{aligned}$$

ETNA
Kent State University
http://etna.math.kent.edu

in which Z_{QR} denotes the new iterate. The higher order and semiseparable plus diagonal cases can be considered in the same way.

The QR algorithm based on the $\vee$ pattern has not been discussed before. However, the idea is a straightforward generalization of the QR algorithm based on the $\wedge$ pattern. Due to the fact that we have switched in some sense the order of both sequences of $n-1$ Givens transformations, we can also switch the interpretation of this algorithm.

We have again two similarity transformations to be performed: $\check{Q}_2^H(\check{Q}_1^H Z \check{Q}_1)\check{Q}_2$. Now, $\check{Q}_1$ is a descending sequence of Givens transformations for expanding the rank structure and $\check{Q}_2$ is an ascending sequence of Givens transformations for removing the newly created rank structure of the intermediate Hessenberg-like matrix.

- The first step can be performed implicitly, similar to the second sequence in the $\wedge$-case. An initial disturbing Givens transformation is applied, followed by $n-2$ structure restoring Givens transformations[6]. As a result we obtain the Hessenberg-like matrix[7]
$$\tilde{Z} = \check{Q}_1^H Z \check{Q}_1.$$
- One can prove that the second step (corresponding to the Givens transformations from bottom to top) can again be seen as performing a step of the QR method without shift on the newly created Hessenberg-like matrix $\tilde{Z}$. After performing the similarity transformation corresponding to $\check{Q}_2$, we obtain the result of performing one step of the QR method without shift applied to the Hessenberg-like matrix $\tilde{Z}$.

NOTE 2.9. *In the similarity transformation related to the $\vee$ pattern, we have that the first step is dependent on the shift μ, whereas the second step is independent of μ. See also Note 2.8 for the iteration related to the $\wedge$ pattern.*

NOTE 2.10. *The remark above makes it clear that this algorithm (as well as the algorithm related to the $\wedge$ pattern) has a kind of contradicting convergence behavior. When we look at the bottom-right corner of the matrix, we have that:*

- *The first step is determined by the shift, and hence creates convergence to the eigenvalue(s) closest to the shift.*
- *The second step corresponds to a QR-step without shift, and hence converges to the smallest eigenvalue(s) in modulus.*

Both convergence behaviors do not necessarily cooperate. In some sense, the second step can damage the improvements made by the first step.

One can opt to remove the second similarity transformation. Unfortunately we will not have a QR factorization and a corresponding QR method anymore. This approach leads to the QH method, which is discussed in Section 4.

Based on the comments above, we would like to use only the factor $\check{Q}_1$ for performing an orthogonal similarity transformation of the matrix Z. As $\check{Q}_1$ is closely related to the QH factorization, a naïve approach would be

$$Z - \mu I = \check{Q}\check{Z},$$

which is a QH factorization of the matrix $Z - \mu I$. We can define the new iteration as

$$Z_{QH} = \check{Q}^H Z \check{Q}.$$

Unfortunately, this creates some problems, as we will see in the next section.

[6]The chasing can be performed in the same way as the chasing step in case of the $\wedge$ pattern.

[7] In Section 4, we will prove that the matrix $\tilde{Z}$ is indeed of Hessenberg-like form.

ETNA
Kent State University
http://etna.math.kent.edu

3. More on the QH factorization and the new QH algorithm. The QH factorization is the basic step in the new QH method. Unfortunately, the QH factorization as proposed above is not properly defined for immediate use in the QH method. We illustrate possible problems with some examples.

EXAMPLE 3.1. *Suppose we have the matrix*

$$Z = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

This matrix is obviously already in Hessenberg-like form. Hence the factorization $Z = IZ$ is a QH factorization. But, in fact, one can apply an arbitrary 2×2 Givens transformation acting on the last two rows, without disturbing the structure. This means that we have an infinite number of QH factorizations for this matrix.

One can also clearly see in Schemes 2.4 and 2.5 that the first three Givens transformations already applied to the matrix create the desired structure. This means that in general one needs $n-2$ Givens transformations to obtain a matrix of the following form (for a 4×4 problem):

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

This matrix is clearly of Hessenberg-like form, and an arbitrary Givens transformation acting on the last two rows can never destroy this rank structure.

NOTE 3.2. *For the higher order case, a similar remark concerning uniqueness can be made. Suppose one has a QH factorization QZ, with Z of $\{p\}$-Hessenberg-like form. One can apply an arbitrary unitary transformation involving the last $p+1$ rows without disturbing the factorization.*

The freedom in constructing the factorization has a direct impact on the QH method, as we can no longer guarantee the preservation of the structure as well as convergence. Later, we will show that we can guarantee this, after having defined our QH factorization in a different essentially unique way.

EXAMPLE 3.3. *Suppose we have the following 3×3 matrix Z and a QH factorization of this matrix. The given matrix Z is clearly a Hessenberg-like matrix, which has its structure preserved under the standard QR algorithm. Let us construct a QH factorization of this matrix:*

$$Z = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & \\ 1 & 0 & \\ & & 1 \end{bmatrix} \begin{bmatrix} 1 & & \\ & 0 & -1 \\ & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} = (\check{G}_1 \check{G}_2)\check{Z} = \check{Q}\check{Z},$$

in which $\check{G}_1\check{G}_2 = \check{Q}$, with G_1 and G_2 two Givens transformations and $\check{Z}$ a Hessenberg-like matrix. Performing the similarity transformation with the unitary matrix Q, we obtain:

$$Z_{QH} = \check{Q}^H Z \check{Q} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}.$$

The new iterate Z_{QH} after a step of the QH method with this factorization is clearly no longer of Hessenberg-like form.

Hence, it is clear that we have to impose some extra constraints on the QH factorization.

ETNA
Kent State University
http://etna.math.kent.edu

Let us consider the following constructive procedure. Suppose that we would like to compute the QH factorization of the matrix $Z+D$. For the Hessenberg-like case, $D=-\mu I$; for the Hessenberg-like plus diagonal case, D incorporates the shift matrix $-\mu I$. Assume all diagonal elements are nonzero. We can write the $\{p\}$-Hessenberg-like matrix Z as follows:

$$Z = RQ,$$

where Q consists of p sequences of Givens transformations. The matrix DQ^H is a $\{p\}$-generalized Hessenberg matrix.

We now obtain

$$\begin{aligned} Z + D &= RQ + DQ^H Q = (R + DQ^H)Q \\ &= \check{Q}\check{R}Q, \end{aligned}$$

where $\check{Q}\check{R} = R + DQ^H$, which is the QR factorization of the left factor in the product. This corresponds to a QH factorization of the original matrix $Z+D$:

$$Z + D = \check{Q}\check{R}Q = \check{Q}\check{Z},$$

with $\check{Z}$ a $\{p\}$-Hessenberg-like matrix. It is important to remark that the matrix $\check{Z} = \check{R}Q$ has exactly the same Q factor in its representation from right to left as the original matrix $Z = RQ$; only the upper triangular matrices $\check{R}$ and R differ. This factorization will be used for the QH method.

DEFINITION 3.4. *A Hessenberg-like matrix Z is said to be irreducible if*

$$\begin{aligned} &\operatorname{rank}(Z(i+1:n, 1:i)) \neq 0, \textit{ for all } i = 1:n-1 \\ &\operatorname{rank}(Z(i:n, 1:i+1)) \ > \ 1, \textit{ for all } i = 1:n-1. \end{aligned}$$

This means that one cannot subdivide the problem, and the low rank structure does not cross the diagonal [18].

In [20], the irreducibility of Hessenberg-like as well as semiseparable matrices is discussed in more detail.

NOTE 3.5. *We now have several remarks:*

- *When considering an irreducible Hessenberg-like matrix Z, one can easily prove uniqueness of the above factorization. Since the matrix Z is irreducible, it has an essentially unique RQ factorization in which all Givens transformations differ from I. This implies that the corresponding Hessenberg matrix H is irreducible, guaranteeing an essentially unique QR factorization of H. Hence, we obtain an essentially unique QH factorization of the matrix Z.*
- *We imposed the constraint that the diagonal elements D needed to be different from zero. In fact, one can without loss of generality also consider zero diagonal elements. This will, however, lead to trivial block divisions in the factorization.*
- *Reconsidering now both examples above, we see that they do not match our constructive procedure.*

DEFINITION 3.6. *The new iteration proposed in this paper is of the following form. Assume a Hessenberg-like plus diagonal matrix $Z+D$ is given and we have a shift μ (with RQ an RQ factorization of Z). Then*

$$\begin{aligned} Z + (D - \mu I) &= RQ + (D - \mu I)Q^H Q \\ &= (R + (D - \mu I)Q^H)Q \\ &= \check{Q}\check{R}Q \\ &= \check{Q}\check{Z}, \end{aligned}$$

ETNA
Kent State University
http://etna.math.kent.edu

which gives us a specific QH factorization of the matrix $Z + D$.

The new iterate is defined as follows

$$\begin{aligned} Z_{QH} + D_{QH} &= \check{Z}\check{Q} + \mu I \\ &= \check{Q}^H (Z + D)\check{Q}. \end{aligned}$$

NOTE 3.7. *We would like to remark that this paper is based on the technical report [24]. The report contains extra material related to the uniqueness of the QH factorization and alternative proofs to predict convergence and preservation of structure. The details are rather technical and we chose not to include them in this paper.*

4. Convergence of the QH method. This method can be considered as a specific case of a more general framework presented in [25]. This framework discusses rational QR iteration steps. In this report, general theoretical convergence results, as well as results on the preservation of structure and so forth, are presented. We will only use the results applicable to our case.

Since the results for the standard Hessenberg-like case are the easiest ones to derive, we will focus attention to this case. The results for Hessenberg-like plus diagonal matrices are more complicated since a diagonal is involved. We will not prove all the details, but state the results.

4.1. A rational QR iteration. Let us interpret the QH iteration in terms of a rational QR iteration. The analysis presented here is similar to the one in [30–32] and is a special case of the rational QR iteration, which was presented in [25].

As discussed in the previous section, the global iteration is

$$\begin{aligned} Z &= RQ, \\ Z + (D - \mu I) &= (R + (D - \mu I)Q^H)Q = \check{Q}\check{R}Q, \\ Z_{QH} + D_{QH} &= \check{Q}^H (Z + D)\check{Q}, \end{aligned}$$

where $Z_{QH} + D_{QH}$ defines the new iterate in the method.

One can rewrite the above formulas and obtain that the matrix $\check{Q}$ is the Q factor in the QR factorization of the matrix product $(Z + (D - \mu I))Z^{-1}$:

$$\begin{aligned} (Z + (D - \mu I))Z^{-1} &= \left(\check{Q}\check{R}Q\right)\left(Q^H R^{-1}\right) \\ &= \check{Q}\check{R}R^{-1}. \end{aligned}$$

This formula illustrates that we have computed the unitary factor of a special function of Z. Depending on the diagonal matrix D, we have to distinguish between two cases: the case in which D is zero, which is the Hessenberg-like case; or the case in which D is an arbitrary diagonal matrix.

4.2. The Hessenberg-like case. In this case the diagonal matrix D equals zero, and μ is a suitably chosen shift. Without loss of generality one can assume Z to be nonsingular, so that the equation above simplifies and we obtain

$$(Z - \mu I)Z^{-1} = \check{Q}\check{R}R^{-1},$$

where $\check{Q}$ is the unitary transformation that will be used to define the new iterate. Since this fits into the framework of rational QR as presented in [25], preservation of structure of the matrix Z follows immediately. This means that the convergence properties of the iteration

ETNA
Kent State University
http://etna.math.kent.edu

performed on the matrix Z are defined by the subspace convergence properties, defined by the rational function $p(\lambda) = (\lambda - \mu)\lambda^{-1}$. These convergence properties, and more advanced results for a general rational iteration of the form $p(\lambda) = (\lambda - \mu)(\lambda - \kappa)^{-1}$, were extensively discussed in [25].

Some initial theoretical results on subspace iteration theory are necessary. Given two subspaces $\mathcal{S}$ and $\mathcal{T}$ in $\mathbb{C}^n$, denote by $P_{\mathcal{S}}$ and $P_{\mathcal{T}}$ the orthonormal projectors onto the subspaces $\mathcal{S}$ and $\mathcal{T}$, respectively. The standard metric between subspaces is defined as

$$d(\mathcal{S}, \mathcal{T}) = \|P_{\mathcal{S}} - P_{\mathcal{T}}\|_2 = \sup_{\substack{\mathbf{s} \in \mathcal{S} \\ \|\mathbf{s}\|_2 = 1}} d(\mathbf{s}, \mathcal{T}) = \sup_{\substack{\mathbf{s} \in \mathcal{S} \\ \|\mathbf{s}\|_2 = 1}} \inf_{\mathbf{t} \in \mathcal{T}} \|\mathbf{s} - \mathbf{t}\|_2$$

if $\dim(\mathcal{S}) = \dim(\mathcal{T})$, and $d(\mathcal{S}, \mathcal{T}) = 1$ otherwise; see [13].

The next theorem states how the distance between subspaces changes when performing subspace iteration with shifted rational functions. The theorem is a generalization of [32, Theorem 5.1].

THEOREM 4.1. *Let $A \in \mathbb{C}^{n \times n}$ be a simple matrix with eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ and associated linearly independent eigenvectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$. Let $V = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n]$ and let κ_V be the condition number of V, with respect to to the spectral*[8] *norm. Let k be an integer $1 \leq k \leq n-1$, and define the invariant subspaces $\mathcal{U} = \langle \mathbf{v}_{k+1}, \ldots, \mathbf{v}_n \rangle$ and $\mathcal{T} = \langle \mathbf{v}_1, \ldots, \mathbf{v}_k \rangle$. Denote by $(p_i)_i$ a sequence of rational functions and let $\hat{p}_i = p_i \ldots p_2 p_1$. Suppose that*

$$\begin{aligned} p_i(\lambda_j) &\neq 0 \quad & j &= 1, \ldots, k, \\ p_i(\lambda_j) &\neq \pm\infty \quad & j &= k+1, \ldots, n, \end{aligned}$$

for all i, and let

$$\hat{r}_i = \frac{\max_{k+1 \leq j \leq n} |\hat{p}_i(\lambda_j)|}{\min_{1 \leq j \leq k} |\hat{p}_i(\lambda_j)|}.$$

Let $\mathcal{S}$ be a k-dimensional subspace of $\mathbb{C}^n$ satisfying

$$\mathcal{S} \cap \mathcal{U} = \{0\}.$$

Let $\mathcal{S}_i = \hat{p}_i(A)\mathcal{S}_0, i = 1, 2, \ldots$, with $\mathcal{S}_0 = \mathcal{S}$. Then there exists a constant C (depending on $\mathcal{S}$) such that for all i,

$$d(\mathcal{S}_i, \mathcal{T}) \leq C\, \kappa_V\, \hat{r}_i.$$

In particular $\mathcal{S}_i \to \mathcal{T}$ if $\hat{r}_i \to 0$. More precisely we have that

$$C = \frac{d(V^{-1}\mathcal{S}, V^{-1}\mathcal{T})}{\sqrt{1 - d(V^{-1}\mathcal{S}, V^{-1}\mathcal{T})}}.$$

The following lemma relates the subspace convergence to the vanishing of certain subblocks of a matrix.

LEMMA 4.2 ([32, Lemma 6.1]). *Suppose $A \in \mathbb{C}^{n \times n}$ is given, and let $\mathcal{T}$ be a subspace that is invariant under A. Assume G to be a nonsingular matrix, and assume $\mathcal{S}$ to be the*

[8] The spectral norm is naturally induced by the $\|.\|_2$ norm on vectors.

ETNA
Kent State University
http://etna.math.kent.edu

subspace spanned by the first k columns of G. (The subspace $\mathcal{S}$ can be considered an approximation of the subspace $\mathcal{T}$.) Assume that $B = G^{-1}AG$, and consider the matrix B, partitioned as

$$B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix},$$

where $B_{21} \in \mathbb{C}^{(n-k)\times k}$. Then we have:

$$\|B_{21}\|_2 \leq 2\,\sqrt{2}\,\mu_G\,\|A\|_2\,d(\mathcal{S},\mathcal{T}),$$

where μ_G denotes the condition number of the matrix G.

For the Hessenberg-like case, the functions are of the form

$$p_i(\lambda) = (\lambda - \mu_i)\lambda^{-1}.$$

Let us compare the convergence behavior of this new iteration to that of the standard QR iteration with shift μ_i. We consider only one iterate, i.e., r_i denotes the contraction rate from step i in the iteration process. For the standard QR algorithm we obtain the contraction ratio

$$r_i^{(QR)} = \frac{\max_{k+1\leq j\leq n}|\lambda_j - \mu_i|}{\min_{1\leq j\leq k}|\lambda_j - \mu_i|}. \tag{4.1}$$

We introduce the constants

$$\omega = \min_{k+1\leq j\leq n}\{|\lambda_j|\},$$
$$\Omega = \max_{1\leq j\leq k}\{|\lambda_j|\}.$$

Calculating now an upper bound for the convergence of the QH method towards the eigenvalue closest to the shift μ_i gives us:

$$r_i^{(QH)} = \max_{k+1\leq j\leq n}\left|\frac{\lambda_j-\mu_i}{\lambda_j}\right| \max_{1\leq j\leq k}\left|\frac{\lambda_j}{\lambda_j-\mu_i}\right| \leq \frac{\Omega}{\omega}\frac{\max_{k+1\leq j\leq n}|\lambda_j-\mu_i|}{\min_{1\leq j\leq k}|\lambda_j-\mu_i|} = \frac{\Omega}{\omega}\,r_i^{(QR)}.$$

This indicates that convergence of the new iteration is comparable (up to a constant) to the convergence of the standard QR method. This constant only creates a small, negligible delay in the convergence. This means that if the traditional QR method converges to an eigenvalue in the lower right corner, the QH method also will converge. Hence, to obtain convergence to a specific eigenvalue λ_j, we choose μ_i close to this eigenvalue. The convergence results prove that this eigenvalue will then be revealed by both the QR and the QH method in the lower right corner.

Moreover, we also have extra convergence, which is not present in the standard QR-case, and which is stems from the factor λ^{-1} in the rational functions.

Define the constants

$$\Delta_i = \max_{k+1\leq j\leq n}\{|\lambda_j - \mu_i|\},$$
$$\delta_i = \min_{1\leq j\leq k}\{|\lambda_j - \mu_i|\}.$$

Similarly to the above, we can define the contraction ratio

$$r_i = \frac{\Delta_i}{\delta_i}\frac{\max_{1\leq j\leq k}|\lambda_j|}{\min_{k+1\leq j\leq n}|\lambda_j|}.$$

ETNA
Kent State University
http://etna.math.kent.edu

Assume now (without loss of generality) that $|\lambda_1| \leq |\lambda_2| \leq \ldots \leq |\lambda_n|$. This means that our convergence rate can be simplified as follows:

$$r_i = \frac{\Delta_i}{\delta_i} \frac{|\lambda_k|}{|\lambda_{k+1}|}.$$

Hence, we get a contraction for all k determined by the ratio λ_k/λ_{k+1}. This is a basic non-shifted subspace iteration taking place for all k at the same time. We remark that this convergence takes place in addition to the convergence imposed by the shift μ_i, which can force, for example, extra convergence towards the bottom-right element.

More information on this specific type of subspace iteration can be found in [25].

4.3. The Hessenberg-like plus diagonal case. The convergence theory related to the Hessenberg-like plus diagonal case is more complicated. In each step of the above method, one will now perform a step of the shifted QR iteration, combined with a nested multishift iteration. The convergence analysis of this method is not so easy compared to the standard QH method for Hessenberg-like matrices. We will not present the global convergence theory, but a brief explanation of the behavior. Similarly to the results in [25, 28], one can derive global convergence results and predictions of the convergence ratios.

We distinguish between two cases. First, we discuss the case in which $\mu = 0$. As we want to compute the specific QH factorization of the matrix $A = Z + D$ in which Z is a Hessenberg-like matrix and D an arbitrary diagonal, we apply the algorithm

$$\begin{aligned} Z &= RQ, \\ Z + D &= (R + DQ^H)Q \\ &= \check{Q}\check{R}Q. \end{aligned}$$

Applying the traditional analysis from above, we obtain

$$(Z + D)Z^{-1} = A(A - D)^{-1} = \check{Q}\check{R}R^{-1}.$$

Hence, we have computed the QR factorization of the original matrix A multiplied by the inverse of A minus a diagonal shift matrix. This diagonal shift creates the nested multishift iteration, with shifts equal to the diagonal elements, similar to the reduction to semiseparable plus diagonal form.

Assuming $\mu \neq 0$, we obtain

$$\begin{aligned} Z &= RQ, \\ Z + D - \mu I &= \left(R + (D - \mu I)Q^H\right) Q \\ &= \check{Q}\check{R}Q. \end{aligned}$$

We also get

$$(Z + D - \mu I)Z^{-1} = (A - \mu I)(A - D)^{-1} = \hat{Q}\hat{R}R^{-1}.$$

This implies that we perform a step of the traditional QR method combined again with the nested multishift iteration.

Hence, in the Hessenberg-like plus diagonal case, we again attain the classical convergence of the QR method plus an extra nested multishift iteration. An interpretation of this kind of subspace iteration and its convergence properties can be found in [28].

NOTE 4.3. *Nothing has yet been mentioned about the preservation of the structure in case of performing this iteration on a Hessenberg-like plus diagonal matrix. Since the QH*

ETNA
Kent State University
http://etna.math.kent.edu

iteration performs a partial QR step related to the $\vee$ pattern as discussed in Subsection 2.3, the preservation of the structure can be derived by modifying the proof of the preservation of the structure in the traditional QR method. The proof can be found in the technical report [24]. We only formulate the theorem.

THEOREM 4.4. *Suppose a Hessenberg-like plus diagonal matrix $Z + D$ is given where $D = \operatorname{diag}([d_1, \ldots, d_n])$, with*

$$Z + D - \mu I = \check{Q}\check{Z},$$

constructed as described above. Then the matrix $\check{Q}^H(Z + D)\check{Q}$ is a Hessenberg-like plus diagonal matrix $Z_{QH} + D_{QH}$, where the diagonal elements of D_{QH} are shifted up one position relative to the diagonal elements of the matrix D, i.e.,

$$D_{QH} = \operatorname{diag}([d_2, \ldots, d_{n-1}, d_n, \beta]),$$

where β is a freely chosen element.

4.4. Summary of QH convergence results. Let us draw some conclusions from this and the previous section. The QH factorization as it was presented initially clearly does not satisfy the needs of an iterative method to compute eigenvalues. For example, the freedom in computing the factorization allowed one to make choices such that the structure was not preserved, making it useless for the design of an eigenvalue solver.

Definition 3.6 provided formulas for computing the factorization in a different way. Based on these relations, we were able to prove that the $\check{Q}$ factor in the QH factorization is actually the unitary factor of the QR factorization of a rational function in the Hessenberg-like (plus diagonal) matrix Z. Hence, all theoretical results for the QH method transform in a certain sense to classical results for (multishift) QR iterations [31, 32].

Being able to use classical results for the (multishift) QR iteration opens several doors. One might, for example, consider the design of an implicit QH method. Standard theorems for constructing implicit algorithms state that the first column of the orthogonal factor, combined with a structure-restoring process applied to the involved matrix, is enough to guarantee that one has performed a step of the QR method on the matrix Z.

Since the $\check{Q}$ factor in the QH-decomposition consists of a descending sequence of Givens transformations, the first column of $\check{Q}$ is only determined by a single Givens transformation. Hence, it is not necessary to follow the complete procedure from Definition 3.6 in order to compute the matrix Q; we only need to determine its first Givens transformation and combine it with a structure-restoring process. This is the subject of the upcoming section.

Both convergence behaviors are very closely related to the convergence behavior in the reduction algorithms to respectively Hessenberg-like and Hessenberg-like plus diagonal form:

- The unitary similarity reduction of an arbitrary matrix to Hessenberg-like form has an extra convergence property compared with the traditional reduction to tridiagonal form. In every step of the reduction process a kind of nested non-shifted subspace iteration also takes place. This nested non-shifted subspace iteration also can be found in the new QH iteration. The standard convergence results for the QR iteration are present, plus an extra subspace iteration convergence; see [15].
- The unitary similarity transformation to Hessenberg-like plus diagonal form has an even more advanced convergence behavior than the reduction to Hessenberg-like form: namely, a nested multishift subspace iteration takes place. A similar phenomenon also takes place in the QH iteration: in every step of the iteration we have the traditional convergence properties plus an extra shifted iteration, which we can see when combining multiple steps as a multishift iteration; see [12, 27, 29].

ETNA
Kent State University
http://etna.math.kent.edu

5. The implicit QH iteration for Hessenberg-like (plus diagonal) matrices. Even though the presented theoretical results might seem complicated, the actual implementation is quite simple, even simpler than the implementation of the QR method.

In this section, we derive an implicit chasing technique for Hessenberg-like plus diagonal matrices. This approach is also valid in the special case of Hessenberg-like matrices, for which the diagonal matrix in the sum is zero.

5.1. An implicit algorithm. In this section, we design an implicit way of performing an iteration of the QH method on a Hessenberg-like plus diagonal matrix.

Based on the results above, we can compute the factorization

$$Z + (D - \mu I) = \check{Q}\check{Z}.$$

The matrix $\check{Q}$ is then used to perform a unitary similarity transformation on $Z + D$:

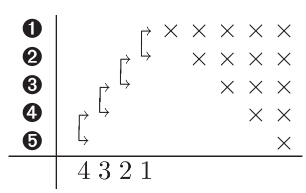
$$Z_{QH} + D_{QH} = \check{Q}^H (Z + D)\check{Q}.$$

The idea of the implicit method is to compute $\check{Q}^H (Z + D)\check{Q}$ based on only the first column of $\check{Q}$ and on the fact that the matrix $Z_{QH} + D_{QH}$ satisfies some structural constraints. This approach is completely similar to the implicit QR-step for tridiagonal/Hessenberg matrices [13, 14] (and also semiseparable matrices [20]).

Because $\check{Q}^H = \check{G}^H_{n-1}\check{G}^H_{n-2} \dots \check{G}^H_1$ consists of a descending sequence of $n-1$ Givens transformations, only the first Givens transformation $\check{G}_1$ is needed to determine the first column of $\check{Q}$. This Givens transformation is applied to the matrix $(Z + D)$, disturbing the Hessenberg-like plus diagonal structure. The remaining $n - 2$ Givens transformations are constructed to restore the structure of the Hessenberg-like matrix, and to obtain $Z_{QH} + D_{QH}$ satisfying Theorem 4.4. After performing these transformations, we know, based on the implicit Q-theorems for Hessenberg-like (plus diagonal) matrices (see [1, 12, 19]), that we have performed a step of the QH method in an implicit manner.

5.2. Assumptions. Before starting the construction of the implicit algorithm we need to assume some things about the Hessenberg-like (plus diagonal) matrix. In the Hessenberg case, one only assumes irreducibility, i.e., the matrix cannot be split up into several subblocks. Here we similarly assume the Hessenberg-like matrix to be irreducible (according to Definition 3.4), and the diagonal minus shift matrix should not have zero elements.

5.3. Computing the initial disturbing Givens transformations. For the actual implementation, we assume the Hessenberg-like matrix Z to be represented by the Givens-vector representation. This can be seen as the QR factorization of the matrix $Z = QR$. We remind the reader that the matrix $Q = G_{n-1}G_{n-2} \dots G_1$ can be factored as a sequence of Givens transformations, where each Givens transformation G_i acts on two successive rows, i and $i + 1$. Graphically, this representation $Z = QR$ is depicted as follows.


(5.1)

The Givens transformations in positions 1 to 4 make up the matrix Q, and the upper triangular matrix R is shown on the right.

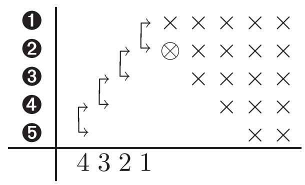
We will now determine a Givens transformation acting on rows 1 and 2 of the matrix $Z+D$ such that the strictly lower triangular rank structure of this matrix also includes the first and the second diagonal element. This is the first Givens transformation needed to compute the QH factorization.

We have

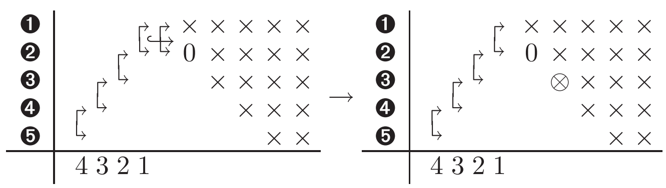
$$Z + (D - \mu I) = QR + (D - \mu I) = Q\,(R + H)\,,$$

where H is a Hessenberg matrix. We now want to apply a sequence of descending Givens transformations to $Z + (D - \mu I)$ so that we obtain a Hessenberg-like matrix $\check{Z}$.

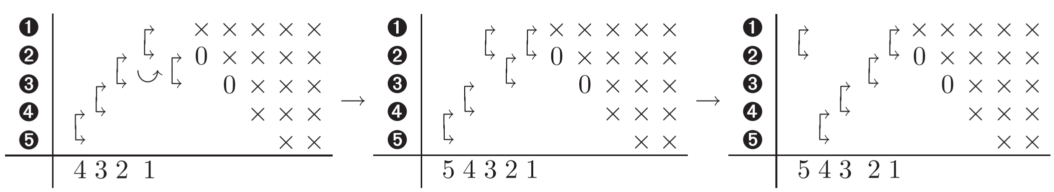
Using the graphical representation we can represent $Q\,(R + H)$ as follows, where the Givens transformations making up Q are shown on the left, and the Hessenberg matrix $R + H$ is shown on the right.



The element marked by $\otimes$ should be annihilated, because we want to obtain a Givens-vector representation of a new Hessenberg-like matrix, namely $\check{Z}$, as in Scheme 5.1. Removing this element by placing a new Givens transformation in position one, and applying the indicated fusion, gives us the following result.



Annihilating the element marked in position $(3, 2)$ by a Givens transformation and performing the shift-through operation at the indicated position, we obtain the following figure.



We remark that the rightmost figure still represents the original matrix $Z + D - \mu I$. Due to the rewriting of the matrix, we can, however, clearly see that performing the Hermitian conjugate of the Givens transformation in position 5 to the left of the matrix $Z + D$ will give a Hessenberg-like structure in the upper left corner of this matrix. This is due to the fact that this upper left part is already represented in the Givens-vector representation.

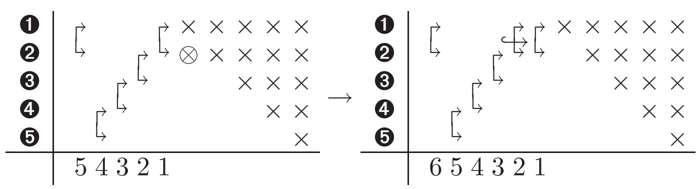
Having calculated this Givens transformation, we can apply it as a similarity transformation to Z, and then, to complete the implicit chasing procedure, restore the structure of this matrix, never again interfering with the first column and row. In the following subsection, we illustrate how to restore the structure of this matrix based on an initial disturbing Givens transformation.

ETNA
Kent State University
http://etna.math.kent.edu

5.4. Restoring the structure. We have a Hessenberg-like plus diagonal matrix $Z + D$ in which $D = \operatorname{diag}([d_1, d_2, \ldots, d_n])$. We know that a step of the QH method results in a Hessenberg-like plus diagonal matrix $Z_{QH} + D_{QH}$ in which $\hat{D} = \operatorname{diag}([d_2, d_3, \ldots, d_n, \beta])$. Assume in the following graphical schemes that all transformations are well-defined.

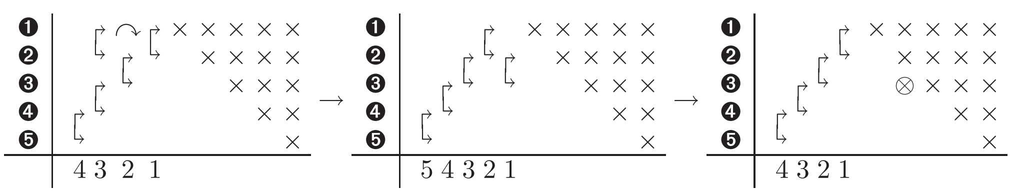
After computing the initial disturbing Givens transformation, we apply this transformation to $Z + D$. Before being able to perform the first transformation we need to rewrite our matrix $Z + D = Z_1 + D_1$, where Z_1 is a Hessenberg-like matrix that differs from Z only in the upper left element, and where $D_1 = \operatorname{diag}([d_2, d_2, d_3, \ldots, d_n])$. Applying the similarity transformation gives us $\check{G}_1^H(Z_1 + D_1)\check{G}_1 = \check{G}_1^H Z_1 \check{G}_1 + D_1$. The diagonal D_1 does not change, because the Givens transformation acts on the first two rows and columns, and the diagonal elements in these positions are both equal to d_2. Our matrix Z_1 can be represented as in Scheme 5.1. After applying the disturbing transformation, this scheme also is disturbed. Then we try to obtain again Scheme 5.1 by applying similarity transformations that do not further affect the first column and row of the matrix.

In the following figures, we do not show the diagonal, but only the effect of the similarity transformation $\check{G}_1$ acting on the matrix Z_1. For simplicity, we assume our matrix to be of size 5×5. Let us write $\tilde{Z}_2 = \check{G}_1^H Z_1 \check{G}_1$.



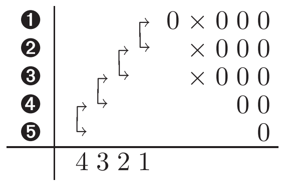
The transformation $\check{G}_1$ applied on the right creates the bulge, marked by $\otimes$ in position $(2, 1)$, whereas the Givens transformation $\check{G}_1^H$ applied on the left can be found in position 5. The bulge marked by $\otimes$ can be annihilated by a Givens transformation as depicted above.

In the following figure, we have combined the Givens transformations in position 1 and 2, by a fusion. We have moved the transformation from position 6 to position 3, and we depicted where to apply the shift-through lemma. The right figure shows the result after applying the shift-through lemma and after creating the bulge, marked with $\otimes$.



We remark once more that the above rearrangements of the Givens transformations did not affect the diagonal matrix D_1. To continue further, we need deal again with D_1.

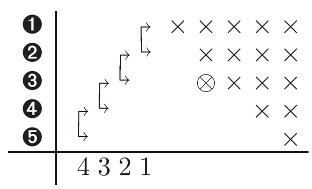
The next similarity Givens transformation acts on columns and rows 2 and 3. To perform the procedure, we first change the diagonal matrix $D_1 = \operatorname{diag}([d_2, d_2, d_3, \ldots, d_n])$ into $D_2 = \operatorname{diag}([d_2, d_3, d_3, \ldots, d_n])$. This change in the diagonal $\tilde{D}_2$ with $D_1 = D_2 + \tilde{D}_2$, and $\tilde{D}_2 = \operatorname{diag}([0, d_2 - d_3, 0, \ldots, 0])$ needs to be incorporated in the scheme above, in the rightmost figure, namely matrix $\tilde{Z}_2$. To incorporate the matrix $\tilde{D}_2$ into $\tilde{Z}_2$, we use the factorization of the matrix $\tilde{Z}_2 = U_2 S_2$ depicted in the rightmost scheme above, where U_2 depicts the combination of the Givens transformations in positions 1 to 4 and S_2 is the upper triangular matrix with the bulge on the right. We obtain that the matrix $\tilde{D}_2 = U_2 U_2^H \tilde{D}_2 = U_2(U_2^H \tilde{D}_2)$ equals the following scheme. The Givens transformations in positions 1 to 4 coincide with U_2 and the sparse matrix on the right equals $(U_2^H \tilde{D}_2)$.



Rewriting all of this into formulas, we obtain

$$
\begin{aligned}
\check{G}_1^H (Z_1 + D_1)\check{G}_1 &= \check{G}_1^H Z_1 \check{G}_1 + D_1 \\
&= \tilde{Z}_2 + D_1 \\
&= \tilde{Z}_2 + \tilde{D}_2 + D_2 \\
&= U_2 S_2 + U_2(U_2^H \tilde{D}_2) + D_2 \\
&= U_2(S_2 + U_2^H \tilde{D}_2) + D_2 \\
&= Z_2 + D_2.
\end{aligned}
$$

It is important that $\tilde{Z}_2$ and Z_2 are factored by the same matrix U_2, and moreover that they have the bulge in exactly the same position. Hence, we can proceed with a similar scheme to the one above, where we now work with Z_2 instead of $\tilde{Z}_2$.



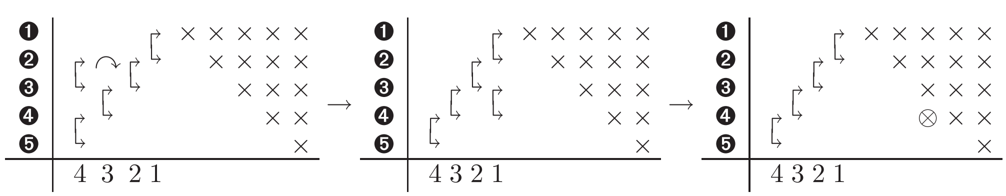
The new scheme looks similar to the one above, but a few elements, including the bulge, have changed.

To continue the implicit procedure, we want to remove the bulge in position $(3, 2)$. In order to do so, we choose a Givens transformation $\check{G}_2$ acting on column 2 and 3, which will remove the bulge. Performing this Givens transformation as a similarity transformation on the matrix $Z_2 + D_2$, we obtain

$$
\begin{aligned}
\check{G}_2^H (Z_2 + D_2)\check{G}_2 &= \check{G}_2^H Z_2 \check{G}_2 + D_2 \\
&= \tilde{Z}_3 + D_2.
\end{aligned}
$$

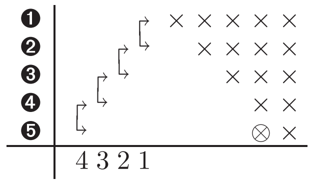
The diagonal D_2 remains unchanged, as the diagonal elements on the second and third positions are equal to each other.

The similarity transformation on Z_2 is schematically depicted as follows:

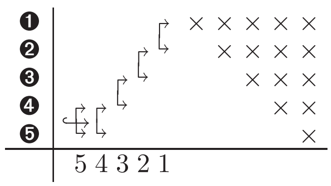


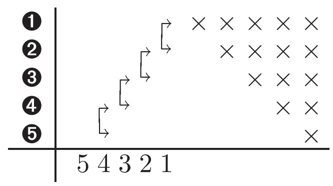
We see that we have now created a new bulge in position $(4, 3)$. A similar technique can now be applied to change the diagonal D_2 to D_3 and to transform $\tilde{Z}_3$ into Z_3. Since the upper triangular parts of the involved matrices are dense, such a chasing step involves $\mathcal{O}(n)$ operations, leading to a global complexity of $\mathcal{O}(n^2)$ for performing one step of the shifted QH method.

We will show only the final step. Assume we have our matrix Z_4 in the following form.



We choose the similarity Givens transformation $\check{G}_4$ to annihilate the element in position $(5, 4)$. Applying this transformation results in the lower left figure. Now, instead of applying the shift-through lemma, we only need to combine the Givens transformations in position 4 and 5, resulting in a Hessenberg-like matrix as we wanted. Moreover, we immediately have the new representation of this Hessenberg-like matrix, and therefore we can immediately perform a new step of the iteration.





The resulting diagonal is $D_5 = \mathrm{diag}([d_2, d_3, \ldots, d_n, \beta])$, where β is freely chosen.

Based on the implicit Q-theorems, we know that we have now implicitly performed a step of the shifted QH method.

6. The QR iteration on Hessenberg matrices is a disguised QH iteration. In the previous part of the paper, we constructed a QH factorization to make the QH method suitable for Hessenberg-like and Hessenberg-like plus diagonal matrices. Let us now compute the QH factorization of a Hessenberg matrix, based on a sequence of descending Givens transformations. We remark that the strictly lower triangular part of a Hessenberg matrix already has semiseparability rank 1. Hence, the descending sequence of Givens transformations is constructed in such a way as to expand the strictly lower triangular rank structure to include the diagonal. Let us first consider the structure of the Givens transformations involved.

COROLLARY 6.1. *Suppose the row $[e, f]$ and the following 2×2 matrix are given*

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}.$$

Then there exists a Givens transformation

$$G = \frac{1}{\sqrt{1+t^2}} \begin{bmatrix} \bar{t} & -1 \\ 1 & t \end{bmatrix}, \tag{6.1}$$

such that the second row of the matrix $G^H A$, and the row $[e, f]$ are linearly dependent. The value of t in the Givens transformation G as in (6.1), is defined as

$$t = \frac{af - be}{cf - de},$$

ETNA
Kent State University
http://etna.math.kent.edu

under the assumption that $cf - de \neq 0$*; otherwise, one may choose* $G = I_2$*.*

Proof. The proof involves straightforward computations. □

Hence, we want to apply a sequence of Givens transformations to the Hessenberg matrix H to obtain the QH factorization. Denote the diagonal elements of the Hessenberg matrix as $[a_1, \dots, a_n]$ and the subdiagonal elements as $[b_1, \dots, b_{n-1}]$. The first Givens transformation acts on rows 1 and 2 and only the first two columns are important, and so, as in the corollary, we consider the matrix

$$A = \left[\begin{array}{cc} a_1 & h_{1,2} \\ b_1 & a_2 \end{array} \right], \tag{6.2}$$

and we want to make the last row dependent of $[0, b_2]$. A Givens transformation with t defined as $t = \frac{a_1 b_2}{b_1 b_2} = \frac{a_1}{b_1}$, is found (assuming b_1 and b_2 to be different from zero). Computing the product $G^H A$ gives us

$$G^H A = \frac{1}{\sqrt{1+t^2}} \left[\begin{array}{cc} \bar{t} & 1 \\ -1 & t \end{array} \right], \left[\begin{array}{cc} a_1 & b_1 \\ b_1 & a_2 \end{array} \right], = \left[\begin{array}{cc} \times & \times \\ 0 & \times \end{array} \right].$$

One can continue this process, and as a result we obtain

$$H = \check{Q}\check{Z} = QR.$$

The Hessenberg-like matrix $\check{Z}$ becomes an upper triangular matrix. Hence, in this case, the QH factorization coincides with the traditional QR factorization, and therefore the QR algorithm for Hessenberg (as well as tridiagonal) matrices also fits into this framework in a certain sense. Better, one can see the QH method as an extension of the traditional QR method.

7. Numerical experiments. In this section, we illustrate the speed and accuracy of the proposed method by various numerical experiments.

7.1. Comparison with the traditional QR method for symmetric semiseparable matrices. In the following experiment, we constructed arbitrary symmetric semiseparable matrices and computed their eigenvalues via the traditional QR method for semiseparable matrices (the implementation from [23] was used). These eigenvalues were compared with those computed by the algorithm described in this paper. Both sets of eigenvalues were compared with the eigenvalues computed by the MATLAB routine `eig`. The following relative error norm was used: denote the vectors containing the eigenvalues as Λ, Λ_{QH}, and Λ_{QR} for respectively `eig`, the QH, and the QR method. The plotted error value, shown in Figure 7.1, equals

$$\frac{\|\Lambda - \Lambda_{QH}\|}{\|\Lambda\|} \text{ and } \frac{\|\Lambda - \Lambda_{QR}\|}{\|\Lambda\|},$$

for both methods. Five experiments were performed, and the line denotes the average accuracy of all five experiments combined. The x-axis denotes the problem sizes, ranging from 100 to 700 in steps of size 50. The cut-off criterion was chosen equal to 10^{-8}. In Figures 7.1 and 7.2 circles denote the results of individual experiments of the QR iteration, whereas stars denote the results for the QH iteration.

Figure 7.2 shows the average number of iterations and the CPU times (in seconds) for both methods. We see that the new method needs, on average, fewer iterations than the QR method.

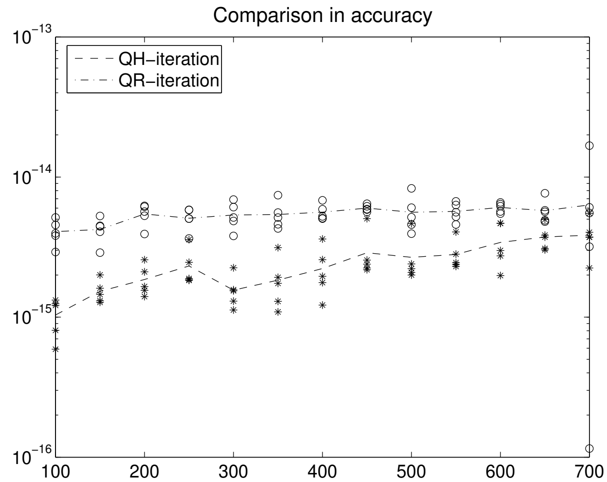


FIGURE 7.1. *Accuracy comparison.*

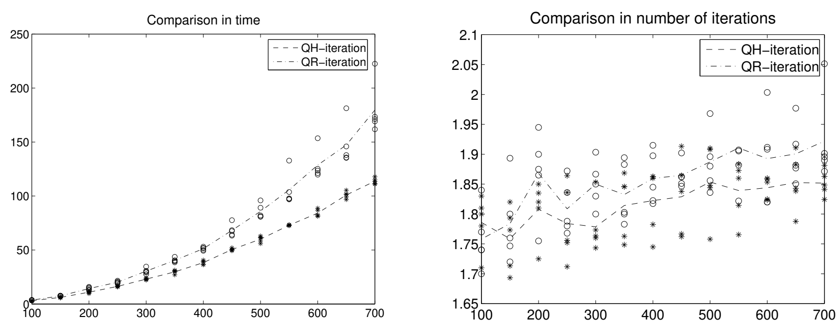


FIGURE 7.2. *CPU times (left) and iteration count (right).*

7.2. Comparison with nonsymmetric complex matrices. In this section, we describe the results of a similar experiment to the one described above, but for complex, not necessarily symmetric, matrices. The examples range from 100 to 700 in steps of size 50, and the cut-off criterion is set to 10^{-14} now.

Figure 7.3 compares the accuracy of the QR and QH methods, and Figure 7.4 shows the average number of iterations and the CPU times (in seconds) for both methods. We see that the new method needs on average much fewer iterations than the QR method.

8. Conclusions. In this paper, we proposed a new method for computing the eigenvalues of Hessenberg-like and Hessenberg-like plus diagonal. The complexity of the methods is half that of the traditional QR methods. Moreover, the new iteration converges in fewer steps than the corresponding QR method.

REFERENCES

[1] R. BEVILACQUA AND G. M. DEL CORSO, *Structural properties of matrix unitary reduction to semiseparable form*, Calcolo, 41 (2004), pp. 177–202.

ETNA
Kent State University
http://etna.math.kent.edu

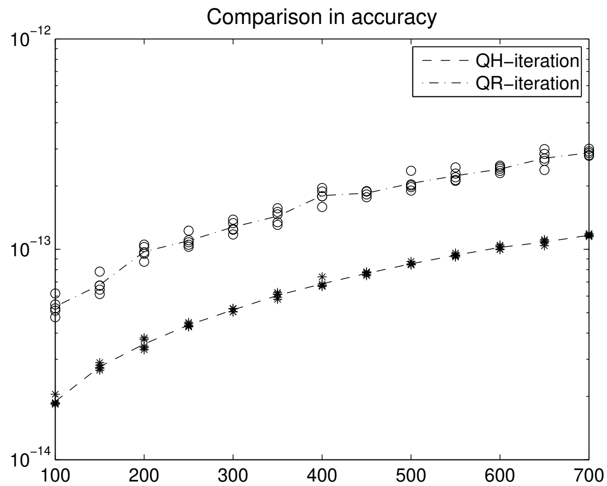


FIGURE 7.3. *Accuracy comparison.*

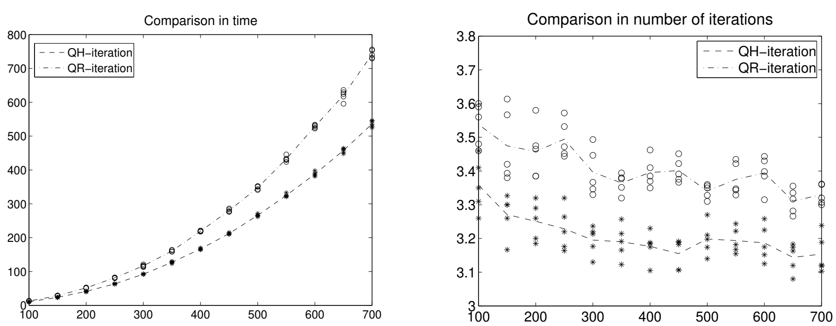


FIGURE 7.4. *CPU times (left) and iteration count (right).*

[2] D. BINDEL, S. CHANDRASEKARAN, J. W. DEMMEL, D. GARMIRE, AND M. GU, *A fast and stable nonsymmetric eigensolver for certain structured matrices*, tech. rep., Department of Computer Science, University of California, Berkeley, California, USA, May 2005.

[3] D. BINDEL, J. W. DEMMEL, W. KAHAN, AND O. A. MARQUES, *On computing Givens rotations reliably and efficiently*, ACM Trans. Math. Software, 28 (2002), pp. 206–238.

[4] D. A. BINI, F. DADDI, AND L. GEMIGNANI, *On the shifted QR iteration applied to companion matrices*, Electron. Trans. Numer. Anal., 18 (2004), pp. 137–152.

[5] 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*, SIAM J. Matrix Anal. Appl., 29 (2007), pp. 566–585.

[6] D. A. BINI, L. GEMIGNANI, AND V. Y. PAN, *Fast and stable QR eigenvalue algorithms for generalized companion matrices and secular equations*, Numer. Math., 100 (2005), pp. 373–408.

[7] S. DELVAUX AND M. VAN BAREL, *The explicit QR-algorithm for rank structured matrices*, Tech. Rep. TW459, Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, 3000 Leuven (Heverlee), Belgium, May 2006.

[8] ———, *A Givens-weight representation for rank structured matrices*, Tech. Rep. TW453, Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, 3000 Leuven (Heverlee), Belgium, Mar. 2006. (To appear in SIMAX).

[9] ———, *A QR-based solver for rank structured matrices*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 464–490.

[10] Y. EIDELMAN AND I. C. GOHBERG, *A modification of the Dewilde-van der Veen method for inversion of*

finite structured matrices, Linear Algebra Appl., 343–344 (2002), pp. 419–450.

[11] Y. EIDELMAN, I. C. GOHBERG, AND V. OLSHEVSKY, *The QR iteration method for Hermitian quasiseparable matrices of an arbitrary order*, Linear Algebra Appl., 404 (2005), pp. 305–324.

[12] D. FASINO, *Rational Krylov matrices and QR-steps on Hermitian diagonal-plus-semiseparable matrices*, Numer. Linear Algebra Appl., 12 (2005), pp. 743–754.

[13] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, Johns Hopkins University Press, Baltimore, Maryland, USA, third ed., 1996.

[14] B. N. PARLETT, *The Symmetric Eigenvalue Problem*, vol. 20 of Classics in Applied Mathematics, SIAM, Philadelphia, Pennsylvania, USA, 1998.

[15] M. VAN BAREL, R. VANDEBRIL, AND N. MASTRONARDI, *An orthogonal similarity reduction of a matrix into semiseparable form*, SIAM J. Matrix Anal. Appl., 27 (2005), pp. 176–197.

[16] E. VAN CAMP, *Diagonal-Plus-Semiseparable Matrices and Their Use in Numerical Linear Algebra*, PhD thesis, Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, 3000 Leuven (Heverlee), Belgium, May 2005.

[17] E. VAN CAMP, N. MASTRONARDI, AND M. VAN BAREL, *Two fast algorithms for solving diagonal-plus-semiseparable linear systems*, J. Comput. Appl. Math., 164–165 (2004), pp. 731–747.

[18] E. VAN CAMP, M. VAN BAREL, R. VANDEBRIL, AND N. MASTRONARDI, *An implicit QR-algorithm for symmetric diagonal-plus-semiseparable matrices*, Tech. Rep. TW419, Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, 3000 Leuven (Heverlee), Belgium, Mar. 2005.

[19] R. VANDEBRIL, M. VAN BAREL, AND N. MASTRONARDI, *An implicit Q theorem for Hessenberg-like matrices*, Mediterr. J. Math., 2 (2005), pp. 259–275.

[20] ———, *An implicit QR-algorithm for symmetric semiseparable matrices*, Numer. Linear Algebra Appl., 12 (2005), pp. 625–658.

[21] ———, *A note on the representation and definition of semiseparable matrices*, Numer. Linear Algebra Appl., 12 (2005), pp. 839–858.

[22] ———, *A parallel QR-factorization/solver of structured rank matrices*, Tech. Rep. TW474, Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, 3000 Leuven (Heverlee), Belgium, Oct. 2006.

[23] ———, *A multiple shift QR-step for structured rank matrices*, Tech. Rep. TW489, Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, 3000 Leuven (Heverlee), Belgium, Apr. 2007.

[24] ———, *A new iteration for computing the eigenvalues of semiseparable (plus diagonal) matrices*, Tech. Rep. TW507, Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, 3000 Leuven (Heverlee), Belgium, Oct. 2007.

[25] ———, *A rational QR-iteration*, Tech. Rep. TW405, Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, 3000 Leuven (Heverlee), Belgium, Sept. 2007.

[26] ———, *Matrix Computations and Semiseparable Matrices, Volume I: Linear Systems*, Johns Hopkins University Press, Baltimore, Maryland, USA, 2008.

[27] 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)*, Tech. Rep. TW436, Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, 3000 Leuven (Heverlee), Belgium, Aug. 2005. (Accepted for publication in Numerische Mathematik).

[28] ———, *On the convergence properties of the orthogonal similarity transformations to tridiagonal and semiseparable (plus diagonal) form*, Numer. Math., 104 (2006), pp. 205–239.

[29] ———, *Orthogonal similarity transformation of a symmetric matrix into a diagonal-plus-semiseparable one with free choice of the diagonal*, Numer. Math., 102 (2006), pp. 709–726.

[30] D. S. WATKINS, *Understanding the QR algorithm*, SIAM Rev., 24 (1982), pp. 427–440.

[31] ———, *QR-like algorithms—an overview of convergence theory and practice*, in The Mathematics of Numerical Analysis, J. Renegar, M. Shub, and S. Smale, eds., vol. 32 of Lectures in Applied Mathematics, American Mathematical Society, Providence, Rhode Island, USA, 1996, pp. 879–893.

[32] D. S. WATKINS AND L. ELSNER, *Convergence of algorithms of decomposition type for the eigenvalue problem*, Linear Algebra Appl., 143 (1991), pp. 19–47.

[33] ———, *Theory of decomposition and bulge-chasing algorithms for the generalized eigenvalue problem*, SIAM J. Matrix Anal. Appl., 15 (1994), pp. 943–967.