



DEFLATING INVARIANT SUBSPACES FOR RANK STRUCTURED PENCILS*

NICOLA MASTRONARDI[†], MARC VAN BAREL[‡], RAF VANDEBRIL[‡], AND PAUL VAN DOOREN[§]

Abstract. It is known that executing a perfect shifted QR step via the implicit QR algorithm may not result in a deflation of the perfect shift. Typically, several steps are required before deflation actually takes place. This deficiency can be remedied by determining the similarity transformation via the associated eigenvector. Similar techniques have been deduced for the QZ algorithm and for the rational QZ algorithm. In this paper, we present a similar approach for executing a perfect shifted QZ step on a general rank structured pencil instead of a specific rank structured one, e.g., a Hessenberg–Hessenberg pencil. For this, we rely on the rank structures present in the transformed matrices. A theoretical framework is presented for dealing with general rank structured pencils and deflating subspaces. We present the corresponding algorithm allowing to deflate simultaneously a block of eigenvalues rather than a single one. We define the level- ρ poles and show that these poles are maintained executing the deflating algorithm. Numerical experiments illustrate the robustness of the presented approach showing the importance of using the improved scaled residual approach.

Key words. Deflating subspace, Rank structured pencil, Perfect shift.

AMS subject classifications. 65F15, 15A18.

1. Introduction. Computing eigenvalues of an $n \times n$ matrix A is one of the most frequently encountered subproblems in numerical linear algebra, and it is not surprising that the celebrated Francis- QR algorithm for computing eigenvalues is in the top 10 list of numerical algorithms of the 20th century [3]. The algorithm is known to be backward stable in the sense that the computed eigenvalues are the exact eigenvalues of a nearby matrix \hat{A} . Moreover, the complexity is cubic in the dimension n of the matrix and is thus of the same order as just solving an $n \times n$ system of equations.

One of the key ingredients of the QR algorithm is the so-called implicit QR step, which operates on an unreduced Hessenberg matrix H and decreases the off-diagonal element $H(n, n-1)$ so that an eigenvalue appears in position $H(n, n)$. If $H(n, n-1) = 0$, then the diagonal element $H(n, n)$ is an eigenvalue (say, λ_0) that can be deflated. In exact arithmetic, such a deflation can be guaranteed when using the eigenvalue λ_0 as a shift in the implicit QR step. But in inexact arithmetic, one often observes a so-called blurring of the perfect shift, and the resulting perturbed off-diagonal element $\hat{H}(n, n-1)$ is not neglectable anymore, precluding a proper deflation of the eigenvalue λ_0 . The reason for this is that the implicit QR step is not

*Received by the editors on August 3, 2023. Accepted for publication on January 17, 2024. Handling Editor: Dario Fasino. Corresponding Author: Raf Vandebril.

The first and fourth authors were partly supported by Gruppo Nazionale Calcolo Scientifico (GNCS) of Istituto Nazionale di Alta Matematica (INdAM). The second author was supported by the Research Council KU Leuven, C1-project C14/17/073 and by the Fund for Scientific Research–Flanders (Belgium), EOS Project no 30468160. The third author was supported by the Research Council KU Leuven (Belgium), project C16/21/002 and by the Fund for Scientific Research – Flanders (Belgium), project G0A9923N. The second and third author were also supported by the Fund for Scientific Research – Flanders (Belgium), project G0B0123N.

[†]Istituto Applicazioni Calcolo, CNR, Via Amendola 122, Bari, 70126, Italy (n.mastronardi@ba.iac.cnr.it).

[‡]Department of Computer Science, KU Leuven, Celestijnenlaan 200A, Heverlee, 3001, Belgium (marc.vanbareel@kuleuven.be, raf.vandebril@kuleuven.be).

[§]Department of Mathematical Engineering, UCLouvain, Av. Lemaitre 4, Louvain-la-Neuve, 1348, Belgium, (vandooren.p@gmail.com).

forward stable and that the lower diagonal elements of the Hessenberg matrix get “blurred” in this process [14]. This is why the deflation of the eigenvalue λ_0 cannot be guaranteed and that several steps are typically required before a deflation actually takes place.

The same phenomenon occurs in the generalized eigenvalue problem $\lambda B - A$ when computing its eigenvalues via the implicit QZ algorithm or the rational QZ algorithm. The implicit QZ step operates on a Hessenberg-triangular pencil of matrices, while the rational QZ step operates on a Hessenberg-Hessenberg pencil of matrices. In both cases, the corresponding implicit QZ steps are backward stable, but forward unstable. The forward instability is the reason why the deflation typically requires several steps before deflation occurs. But in many applications, the deflation of a particular eigenvalue is of crucial importance. The reconstruction of the Jordan structure of an eigenvalue of a matrix A [5, 8], the computation of the index of a Differential-Algebraic system of equations [6], and the removal of an interpolation point in a reduced order scheme for numerical integration [7] are typical examples where such deflations play a crucial role.

It was shown in the papers [5, 8, 6, 7] that the forward instability of the implicit QR -like steps can be avoided by first computing the eigenvector corresponding to the perfect shift accurately enough by the improved scaled residual approach and use it to construct the sequence of Givens transformations that perform the QR -like step. Similar results were also deduced for the QZ algorithm and the rational QZ algorithm. It should be pointed out that in those papers, the analysis was performed on the backward QR step rather than the forward one. This means the deflated eigenvalue appears on top of the matrix, rather than at the bottom, but the conclusions are the same.

In this paper, we present a similar approach for executing a perfect implicit shifted QR step, relying on the rank structures present in the matrix pencil. There is a vast literature on QR algorithms for rank structured matrices, see e.g., the books [12, 1, 4] and the references therein. The approach in this article is, however, significantly different as not all, but only a particular eigenvalue needs to be deflated; as such this paper generalizes the results of [8] to sparsity patterns that are more general than Hessenberg matrices. A theoretical framework is presented for dealing with general rank structured pencils and deflating subspaces. This allows to deflate a block of eigenvalues instead of a single one. The rank structured approach developed in this paper can be viewed as a generalization of the multishift ideas developed in [2, 9]. The numerical experiments illustrate the robustness of the presented approach.

2. Notation. In this paper, we will consider $n \times n$ regular matrix pencils. An $n \times n$ pencil $\lambda B - A$ is said to be regular if $\det(\lambda B - A)$ is not identically zero, which implies that its set of eigenvalues, denoted as $\Lambda(B, A) := \{\lambda_1, \dots, \lambda_n\}$, is a self-conjugate set when the pencil is real. The finite eigenvalues are the roots of the polynomial $\det(\lambda B - A)$. When B is singular this pencil also has infinite eigenvalues. For this reason, it is preferred to define the eigenvalues as the ratios $\lambda_i := a_i/b_i$ where the pairs (a_i, b_i) satisfy the homogeneous equation $\det(aB - bA) = 0$. Since these pairs are only defined up to a common scalar factor, they are typically constrained to satisfy $|a_i|^2 + |b_i|^2 = 1$. The so-called chordal distance $\chi(\lambda_i, \tilde{\lambda}_i)$ between an eigenvalue $\lambda_i = a_i/b_i$ and its perturbed version $\tilde{\lambda}_i = \tilde{a}_i/\tilde{b}_i$ can then be expressed simply in terms of the scaled pairs:

$$\chi(\lambda_i, \tilde{\lambda}_i) := \frac{|\lambda_i - \tilde{\lambda}_i|}{\sqrt{1 + |\lambda_i|^2} \sqrt{1 + |\tilde{\lambda}_i|^2}} = |a_i \tilde{b}_i - b_i \tilde{a}_i|.$$

An m -dimensional deflating subspace of a regular pencil $\lambda B - A$ is defined as any subspace \mathcal{S} such that $\dim(AS + BS) = \dim \mathcal{S} = m$. An equivalent condition is the existence of a $m \times m$ regular pencil $\lambda \hat{B} - \hat{A}$ and $n \times m$ matrices S and U such that

$$(2.1) \quad (\lambda B - A)S = U(\lambda \hat{B} - \hat{A}),$$

which implies that $\mathcal{S} = \text{Im}S$ and $\mathcal{U} := AS + BS = \text{Im}U$. The pair $(\mathcal{U}, \mathcal{S})$ is then called a pair of deflating subspaces. The spectrum $\Lambda(\hat{B}, \hat{A})$ of the associated pencil $\lambda \hat{B} - \hat{A}$ is then a subset of the spectrum of $\lambda B - A$ [10], and if $\Lambda(\hat{B}, \hat{A})$ is self-conjugate, then the equation (2.1) is real. In this paper, we will focus on the complex case, but all the results can be shown to extend to the real case as well when the spectra of the pencils $\lambda B - A$ and $\lambda \hat{B} - \hat{A}$ are self-conjugate.

Matrices and submatrices are denoted by capital letters, i.e., A, B, H . The entry (i, j) of the matrix M is denoted by $M(i, j)$, and submatrices are denoted by the colon notation of `Matlab`: $M(i : j, k : l)$ denotes the submatrix of M formed by the intersection of rows i to j and columns k to l , and $M(i : j, :)$ and $M(:, k : l)$ denote the rows of M from i to j and the columns of M and from k to l , respectively. The identity matrix of order n is denoted by I_n or, if there is no ambiguity, simply by I . Its first column is denoted by e_1 . Generic entries different from zero in matrices or vectors are denoted by “ \times .” The machine precision is denoted by ϵ_M . A so-called *phase matrix* is a diagonal matrix of elements $e^{i\phi_k}$ which are unit modulus complex numbers. We denote a (complex) Givens rotation between two adjacent rows or columns i and $i + 1$, by b

$$G_i = \begin{bmatrix} I_{i-1} & & & \\ & c & s & \\ & -\bar{s} & c & \\ & & & I_{n-i-1} \end{bmatrix}, \quad \begin{bmatrix} c & s \\ -\bar{s} & c \end{bmatrix} \begin{bmatrix} c & s \\ -\bar{s} & c \end{bmatrix}^H = I_2$$

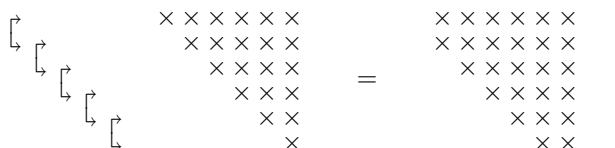
with c real and s complex valued.

3. Rank structured matrices. We introduce rank structured matrices as matrices having a particular QR factorization. Next we introduce patterns of rotations and manners of operating on rotations.

3.1. Rank structured matrices and the QR factorization. The presented algorithm can be described most easily with the notation used extensively in the monographs of Vandebril, Van Barel, and Mastronardi [11, 12]. In this section, we reconsider the essentials needed for this article, but more details on structured rank matrices and using rotations to operate on them, can be found in these monographs.

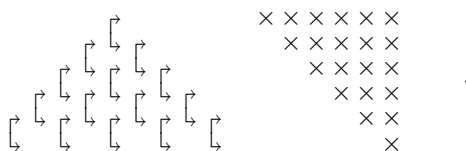
We will describe the new algorithm in a graphical fashion by operating directly on patterns of rotations.

A *rotation* is denoted by a bracket \curvearrowright , where the arrows point at the rows that will be affected when executing the rotation from the left. When having several rotations acting on a matrix, the combination of these rotations forms a *pattern* of rotations. Consider for example as pattern, a descending sequence of rotations. The order in which the rotations are applied matters: the rotation closest to the matrix has to be executed first. We leave it to the reader to verify that the result of applying all the rotations to the upper triangular matrix results in a Hessenberg matrix:



The factorization $QR = A$, where Q is an ascending sequence of rotations, results in a Hessenberg-like matrix A . This matrix is of structured rank form: all its submatrices $A(i : n, 1 : i)$ taken out of the part below and including the diagonal are of rank at most one.

For a general dense matrix, computing the QR decomposition results in a pyramidal pattern of rotations¹:

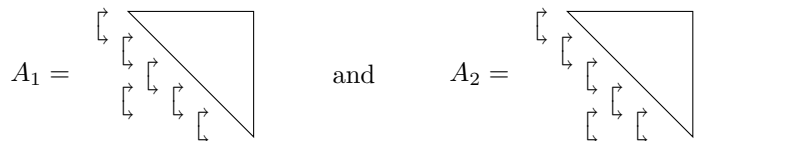


In this paper, we consider the following generic definition of a rank structured matrix.

DEFINITION 3.1 (Rank Structured Matrix). *A matrix A is said to be rank structured if, in the pyramidal pattern of rotations resulting from factoring the matrix Q into rotations, one or more rotations are missing.*

Note that performing matrix computations with rank structured matrices is more efficient when more rotations are missing.

For instance, the matrices below are rank structured. More precisely, we have two Hessenberg matrices, both having an extra bulge. In the pictorial representation, we used the triangle as a shorthand notation for an upper triangular matrix:



Two matrices are said to have the same rank structure if they admit an identical pattern in the factorization of the matrix Q in their QR factorization. For example, for an irreducible Hessenberg matrix the pyramidal shape will always be a descending sequence of rotations. Reconsidering the two matrices above, we see that they have a different rank structure as the bulge appears at a different spot. Consider however the projected matrices $[I_5 \ 0]A_1[I_5 \ 0]^T$ and $[0 \ I_5]A_2[0 \ I_5]^T$, then we observe that both matrices have the same rank structure. To operate on, and modify the patterns of rotations, we need to be able to manipulate them.

3.2. Manipulating rotations. To prove correctness of the algorithm, we will rely on simple manipulations with rotations. Two essential building blocks are the basic operations known as *fusion* and *turnover* (see the monographs of Vandebril, Van Barel, and Mastronardi [11, 12] for more details).

¹We note that a pattern is not unique, different factorizations of Q may exist.

The product of two rotations is again a rotation, which we call a **fusion** of two rotations. We can denote this pictorially as:

$$\begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} = \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} .$$

The product of three rotations, as depicted on the left of the equation below, can be refactored as the product of three rotations, depicted on the right here below:

$$\begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} = \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} .$$

This operation where the pattern is turned upside-down is called a **turnover**.

Suppose now that we have a particular pattern of rotations, say for example, of the following form:

$$\begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} .$$

The turnover is then not applied individually, but several times in a row to allow interactions between such groups of rotations. Suppose, for example, we position the above pattern of rotations on the left of another pattern of rotations, namely an ascending one, then we can apply the turnover operation several times to move the entire pattern of rotations on the left of the ascending sequence to the right of this sequence. We obtain the following:

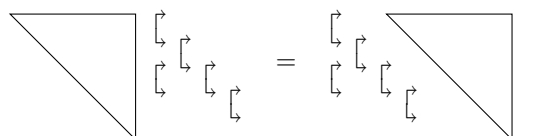
$$\begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} = \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} \begin{array}{|c|} \hline \rightarrow \\ \hline \end{array} .$$

We see that the pattern on the left of the ascending sequence is roughly identical to the one on the right of it. The big difference is that it has shifted down one position. This shows that one can transfer a pattern of rotations from left to right through an ascending sequence, thereby moving it down a row.

It is worth noting that the pattern that we moved down was not acting on the last row. It is, however, perfectly possible that there is a rotation acting on rows n and $n - 1$. This rotation will then be absorbed by a fusion with the bottom rotation of the ascending sequence, so that this rotation will simply disappear. We conclude that the upper part of a pattern of rotations is preserved when passing through an ascending sequence of rotations.

Next to manipulating the inter-position of rotations, we also need to interact with upper triangular matrices. If we have the product of a pattern of rotations with an upper triangular and the pattern is positioned on the right of the upper triangular matrix, we can factor again the product in such a way that

the pattern reappears on the left of an upper triangular matrix [13, Lemma 2.6]. Clearly, all actors, which are the rotations and the upper triangular matrix, change, but the pattern itself remains unaltered. Pictorially, we have the following:



We are now ready to describe the algorithms to deflate subspaces.

4. Perfect shifts and deflating subspaces. We generalize the concept of a perfect shift to a deflating subspace. First, we discuss the Hessenberg-Hessenberg pencil, followed by a generic rank structured pencil. The pencil can have complex values. In the numerical experiments of Section 7, we are working with real pencils using real arithmetic where the deflating subspace is connected to a real eigenvalue or two complex conjugate eigenvalues.

4.1. Perfect shifts for deflating subspaces of Hessenberg-Hessenberg pencils. We draw from previous work by Mastronardi, Van Barel, Vandebril, and Van Dooren [7] on the perfect single and double shifted framework for Hessenberg-Hessenberg (HH) pencils, which arise naturally in the context of rational Krylov subspaces. The perfect double-shift case is essential in retaining real arithmetic, when the original pencil is real. We can, however, extend this to complex deflating subspaces.

THEOREM 4.1. *Let $H - \lambda K$ be a regular HH pencil, where $\lambda_1, \dots, \lambda_m$ are some generalized eigenvalues of the pencil, let $\lambda_i = a_i/b_i$ with $|a_i|^2 + |b_i|^2 = 1$, for each $i = 1, \dots, m$. Suppose there exists a pair of deflating subspaces $(\mathcal{U}, \mathcal{S})$ of dimension m with bases U and S , such that $(bH - aK)S = U(b\tilde{H}_{11} - a\tilde{K}_{11})$, and the generalized eigenvalues of $b\tilde{H}_{11} - a\tilde{K}_{11}$ equal $\lambda_1, \dots, \lambda_m$. Then, the following holds.*

1. Consider a unitary matrix Q yielding a QR factorization $S = Q^H R$. This matrix Q can be decomposed in sequences of rotations $G^{(r,i)} = G_i^{(r,i)} \cdots G_{n-1}^{(r,i)}$, for $i = 1, \dots, m$, where $Q = G^{(r,1)} \cdots G^{(r,m)}$.
2. There is a matrix Z composed of m sequences of rotations

$$G^{(\ell,i)} = G_i^{(\ell,i)} \cdots G_{n-1}^{(\ell,i)}, \quad i = 1, \dots, m, \quad \text{and} \quad Z = G^{(\ell,1)} \cdots G^{(\ell,m)},$$

such that the triple (H, K, S) , is transformed into an equivalent one

$$(\hat{H}, \hat{K}, \hat{S}) := (ZHQ^H, ZKQ^H, QS),$$

such that QS is upper triangular and the eigenvalues of $[I_m, 0](\hat{H} - \lambda\hat{K})[I_m, 0]^T$ equal $\lambda_1, \dots, \lambda_m$ and $[0, I_{n-m}](\hat{H} - \lambda\hat{K})[0, I_{n-m}]^T$ is in HH form.

Proof. Building on the results from [11] and [12], we know how to compute the QR factorization of S , via several sequences of rotations. The first sequence $G^{(r,1)} = G_1^{(r,1)}, \dots, G_{n-1}^{(r,1)}$ creates zeros in the first column in entries $2, \dots, n$. The second sequence $G^{(r,2)} = G_2^{(r,2)}, \dots, G_{n-1}^{(r,2)}$ creates zeros in the second column in entries $3, \dots, n$, and so forth. A proof along the same lines as in [7] can be formulated, but in the remainder of the text we formulate a more constructive approach to prove correctness. We illustrate that applying these transformations to the right of the pencil creates a subblock of low rank that can be removed with an identical number of sequences of rotations from the left.

Though not mentioned explicitly in the theorem above, there is something particular happening with the poles. The poles of an HH pencil are defined as the ratio between the subdiagonal elements of the pencil matrices. After a perfect shift step, the poles will move down one position on the diagonal. Something similar happens here, since the poles will move down m positions.

The positioning of the poles could have been mentioned in the theorem as well, since the rational QZ algorithm is based on this property. But for the purpose of our algorithm, the positioning of the poles is not important. We will prove this property in Section 6 which implies the theoretical equivalence of executing a perfect shift step with the rational QZ algorithm.

4.2. Perfect shifts for deflating subspaces of rank structured pencils. Theorem 4.1 can be extended to general rank structured pencils, for which we will use the shorthand notation RS pencil.

THEOREM 4.2. *Let $H - \lambda K$ be a regular RS pencil, where $\lambda_1, \dots, \lambda_m$ are some generalized eigenvalues of the pencil, say $\lambda_i = a_i/b_i$, with $|a_i|^2 + |b_i|^2 = 1$, for each $i = 1, \dots, m$. Suppose there exists a pair of deflating subspaces $(\mathcal{U}, \mathcal{S})$ of dimension m with bases U and S , such that $(bH - aK)S = U(b\tilde{H}_{11} - a\tilde{K}_{11})$, and the generalized eigenvalues of $b\tilde{H}_{11} - a\tilde{K}_{11}$ equal $\lambda_1, \dots, \lambda_m$. Then, the following holds.*

1. *There exists a matrix Q yielding the QR factorization $S = Q^H R$. This matrix Q can be decomposed in sequences of rotations $G^{(r,i)} = G_i^{(r,i)} \cdots G_{n-1}^{(r,i)}$, for $i = 1, \dots, m$, with $Q = G^{(r,1)} \cdots G^{(r,m)}$.*
2. *There is a matrix Z composed of m sequences of rotations*

$$G^{(\ell,i)} = G_i^{(\ell,i)} \cdots G_{n-1}^{(\ell,i)}, \quad i = 1, \dots, m, \quad \text{and} \quad Z = G^{(\ell,1)} \cdots G^{(\ell,m)},$$

such that the triple (H, K, S) , is transformed into an equivalent one

$$(\hat{H}, \hat{K}, \hat{S}) := (ZHQ^H, ZKQ^H, QS),$$

such that QS is upper triangular and the eigenvalues of $[I_m, 0](\hat{H} - \lambda\hat{K})[I_m, 0]^T$ equal $\lambda_1, \dots, \lambda_m$. The pencil $[0, I_{n-m}](\hat{H} - \lambda\hat{K})[0, I_{n-m}]^T$ has the same rank structure as the pencil $[I_{n-m} 0](H - \lambda K)[I_{n-m} 0]^T$, which means that the rank structure has moved down m positions.

A constructive sketch of the proof is provided further in the text.

Just like for the HH pencil, the poles will move down in this setting. However, we do need a more general definition on the poles of an RS pencil as the elements might extend beyond the subdiagonal. In case of a block HH pencil, which has elements below the subdiagonal, the poles are considered as eigenvalues linked to those blocks. In essence, the poles are the eigenvalues of the pole pencil $H_p - \lambda K_p$ where,

$$H_p - \lambda K_p := \begin{bmatrix} 0 & I_{n-1} \end{bmatrix} (H - \lambda K) \begin{bmatrix} I_{n-1} \\ 0 \end{bmatrix}.$$

When the pole pencil is upper triangular, the poles are the ratios of the diagonal elements of H_p and K_p . Clearly in this case, they obey a particular ordering, but also in the block case the blocks are clearly ordered. After a single shifted rational QR step, the ordering will not change, but the poles or blocks of poles will all move down one position along the diagonal.

For a generic pencil, we can define different levels of poles. They extend the level-1 poles that were described above.

DEFINITION 4.3. *The level- ρ poles are defined as the eigenvalues of the pencil:*

$$H_{p_\rho} - \lambda K_{p_\rho} := \begin{bmatrix} 0 & I_{n-\rho} \end{bmatrix} (H - \lambda K) \begin{bmatrix} I_{n-\rho} \\ 0 \end{bmatrix}.$$

This definition might lead to singular pole pencils. If we encounter a $0/0$ pole, we will call it undefined and denote it by \aleph . We have thus complex poles, but poles can also be infinite or undefined. Again, we will see that when the level- ρ pole pencil reveals a particular ordering of the level- ρ poles this ordering will be maintained and will shift down.

THEOREM 4.4. *In Theorem 4.2, we can also state that all level- ρ poles of $[0, I_{n-m}](\hat{H} - \lambda\hat{K})[0, I_{n-m}]^T$ equal those of $[I_{n-m}, 0](H - \lambda K)[I_{n-m}, 0]^T$ and moreover, the ordering is preserved.*

The proof of this theorem is sketched in Section 6.

5. The algorithm and proof of correctness.

In Subsection 5.1 we give an algorithm to deflate deflating subspaces for rank structured pencils. In subsection 5.2, we prove that this algorithm preserves the structure shifted down along the diagonal.

5.1. The algorithm. The proposed algorithm is essentially simple in nature and builds on Theorems 4.1 and 4.2. It consists of four steps:

1. Construct the matrix Q as factored in m sequences of rotations, each sequence reducing a column in S .
2. Apply these transformations to the right of the pencil $H - \lambda K$.
3. Construct the matrix Z as factored in m sequences of rotations, each sequence reducing one of the first m columns of H or K . This matrix Z can be constructed equivalently based on H or K .
4. Apply the transformation matrix Z to the left of $H - \lambda K$.

Two observations are in place

- The actual implementation does not necessarily execute the four steps in the order described. Applying the full matrix Q does not necessarily need to take place before Z can be constructed; determining Z is not entirely depending on the full application of Q .
 As soon as the necessary information in the bottom rows of the pencil $H - \lambda K$ resulting from applying Q is available to determine Z , one can already start with constructing and applying Z . This technique will be used in Section 6.
- In step 3, we choose to reduce either H or K , theoretically it does not matter which one is chosen. However, for reasons of numerical stability, we always determine the transformation on the largest elements.

5.2. Proof of structure preservation and deflation. In this section, we prove that the described algorithm will allow us to deflate a submatrix with prescribed eigenvalues.

The rank structure in the lower triangular part of H and K is revealed by the pattern of rotations in the matrices Q_H and Q_K , with $H = Q_H R_H$ and $K = Q_K R_K$ the QR factorizations of H and K , respectively. The analysis presented here is general in the sense that the rank structure of H and K can be different, e.g., H is Hessenberg-like and K is Hessenberg with a bulge. We will use this setting as an example to illustrate the flow of the proof.

So, our example pencil matrices admit the following factorization:



Suppose the matrix pencil (H, K) has a deflating subspace of dimension m , with basis vectors the columns of S , i.e.,

$$HS - KSA = 0,$$

where Λ is bounded (i.e., there are no infinite eigenvalues), and S has rank m .

1. The first step of the algorithm is folding up the matrix S by m sequences of rotations. This results in a unitary matrix Q such that QS becomes upper triangular. We denote this matrix Q as Q_1 ; as a result, we get $Q_1S = R_S$.

In our example, we keep things simple and assume $m = 1$, as a consequence folding up the single vector S consists of a single descending sequence of core transformations:

$$\begin{matrix} \curvearrowright & & & & \\ & \curvearrowright & & & \\ & & \curvearrowright & & \\ & & & \curvearrowright & \\ & & & & \curvearrowright \end{matrix} \begin{matrix} | \\ | \\ | \\ | \\ | \end{matrix} = \sigma e_1, \quad |\sigma| = \|S\|_2.$$

Note that in the general case, folding up S , to make it upper triangular, will lead to m descending sequences of rotations, of decreasing length.

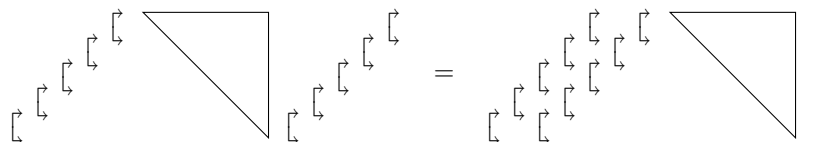
2. In the second step, we pass the unitary matrix Q_1^H through the triangular parts R_H and R_K of H and K from the right to the left resulting in unitary matrices $Q_{2,H}$ and $Q_{2,K}$ where the resulting pattern of rotations is identical but the individual rotations can be different. Mathematically we get the following formulas

$$\begin{aligned} 0 &= HQ_1^H Q_1 S - KQ_1^H Q_1 S \Lambda \\ &= Q_H (R_H Q_1^H) R_S - Q_K (R_K Q_1^H) R_S \Lambda \\ &= Q_H (Q_{2,H} \hat{R}_H) R_S - Q_K (Q_{2,K} \hat{R}_K) R_S \Lambda. \end{aligned}$$

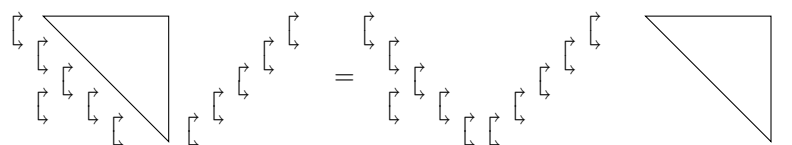
In our example, Q_1^H is the Hermitian conjugate of a descending sequence and becomes thus ascending:

$$Q_1^H = \begin{matrix} & & & & \curvearrowright \\ & & & & & \curvearrowright \\ & & & & & & \curvearrowright \\ & & & & & & & \curvearrowright \\ & & & & & & & & \curvearrowright \\ \curvearrowright & & & & & & & & \end{matrix}.$$

Let us take a closer look at the structure of the matrices involved in our specific example. The matrix HQ_1^H in factored form $Q_H R_H Q_1^H$ looks like the left-hand side of the equation below. On the right-hand side we see the factorization $Q_H Q_{2,H} \hat{R}_H$ after passing the sequence on the right through the upper triangular matrix:



A similar way of reasoning can be applied to the Hessenberg matrix with bulge K . On the left-hand side, we see $Q_K R_K Q_1^H$, on the right-hand side $Q_K Q_{2,K} \hat{R}_K$:

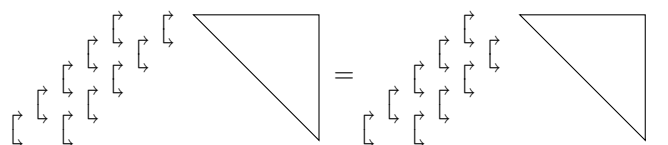


3. In the third step we move the entire pattern of rotations, originating from the QR factorizations of K and H through the ascending sequences of rotations. As a consequence the patterns will move down. Shifting the pattern Q_H to the right of $Q_{2,H}$ leads to $Q_{3,H}$ and correspondingly shifting the pattern Q_K to the right of $Q_{2,K}$ leads to $Q_{3,K}$. As both $Q_{2,H}$ and $Q_{2,K}$ are composed of m ascending sequences of rotations the patterns in Q_H and Q_K will move down m places. We end up with

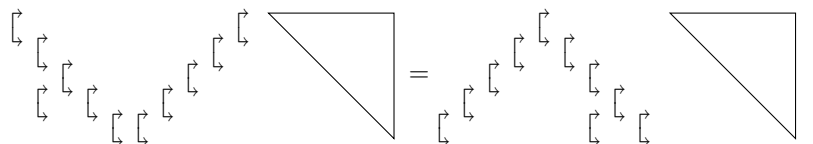
$$\begin{aligned}
 0 &= (Q_H Q_{2,H}) \hat{R}_H R_S - (Q_K Q_{2,K}) \hat{R}_K R_S \Lambda \\
 &= (Q_{3,H} \hat{Q}_H) \hat{R}_H R_S - (Q_{3,K} \hat{Q}_K) \hat{R}_K R_S \Lambda.
 \end{aligned}$$

Both $Q_{3,K}$ and $Q_{3,H}$ still consist of m ascending sequences of rotations. Looking at the matrices \hat{Q}_H and \hat{Q}_K we see that the rank structured pattern of the original matrices H and K has moved down m places.

For our example, we only have a single ascending sequence. Note that moving the rotations from the left to the right of the ascending sequence will move them down a single position, and we dispose of the last rotation. The factorization involving the matrix H looks as follows



Similarly, we get for the factorization involved with the matrix K



4. The previous item revealed the preservation of the rank structure. It remains to prove that there exist m sequences of rotations that allow us to transform the factorizations $Q_{3,H} \hat{Q}_H \hat{R}_H$ and $Q_{3,K} \hat{Q}_K \hat{R}_K$ simultaneously to block diagonal form, such that the upper left $m \times m$ block can be deflated. As the patterns of rotations in Q_H and Q_K have moved down m positions, we see that the first m columns

of $Q_{3,H}\hat{Q}_H\hat{R}_H$ and $Q_{3,K}\hat{Q}_K\hat{R}_K$ are determined by $Q_{3,H}$ and $Q_{3,K}$ only and do not depend on \hat{Q}_H nor on \hat{Q}_K . We get

$$(5.2) \quad Q_{3,H}\hat{Q}_H\hat{R}_H[I_m, 0]^T = Q_{3,H}\hat{R}_H[I_m, 0]^T$$

$$(5.3) \quad Q_{3,K}\hat{Q}_K\hat{R}_K[I_m, 0]^T = Q_{3,K}\hat{R}_K[I_m, 0]^T.$$

In the next step, we multiply the equation

$$(5.4) \quad 0 = Q_{3,H}\hat{Q}_H\hat{R}_H R_S - Q_{3,K}\hat{Q}_K\hat{R}_K R_S \Lambda,$$

on the left with the Hermitian conjugate of $Q_{3,H}$. This corresponds to bringing the first m columns in upper triangular form (see Equation (5.4)), which is actually how we determine $Q_{3,H}$ in practice. As a result we get

$$0 = \hat{Q}_H\hat{R}_H R_S - Q_{3,H}^H Q_{3,K}\hat{Q}_K\hat{R}_K R_S \Lambda.$$

Because of Equations (5.2) and (5.3), we get that the first m columns of $Q_{3,H}^H Q_{3,K}\hat{Q}_K\hat{R}_K R_S \Lambda$ must necessarily be in upper triangular form as well. As a result, $Q_{3,H}$ and $Q_{3,K}$ must be essentially identical (differing only up to some phase factors). This proves that the upper left $m \times m$ block of the transformed pencil is now in upper triangular form and allows to be deflated easily.

In fact, we multiplied Equation (5.4) on the left with the Hermitian conjugate of $Q_{3,H}$, for reasons of numerical stability we can take $Q_{3,K}$, without changing the result of the algorithm.

Let us review our example once more. Folding up the first column of either the factorization involving K or H is equivalent to simply deleting the ascending sequence of rotations in both matrices (taking care of possible phase factor differences). As a consequence, we end with the factorization involving the matrix H on the left, and the one involving K on the right:



Obviously, we have the upper left 1×1 part of the pencil in upper triangular form.

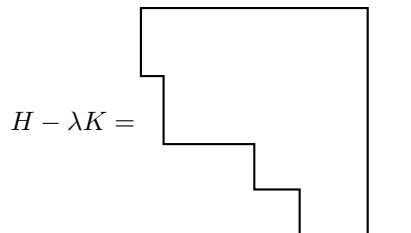
5. After having executed these transformations, we can obviously deflate the upper left $m \times m$ block, thereby retaining a novel smaller RS pencil.

The procedure above proved that the algorithm allows to deflate the upper left $m \times m$ block, and also retains the structured rank of the matrices involved in the pencil.

6. Level- ρ poles are maintained. In this section, we will prove, by induction, that all level- ρ poles are preserved. This observation could possibly be used, e.g., to enhance the numerical stability of the deflating algorithm. We will not consider generic RS pencils, but only block Hessenberg pencils. We remark that every RS pencil can be transformed via transformation on the left to a block Hessenberg pencil. Removing the low rank parts in one matrix will create a block structure in the other matrix. We will therefore restrict ourselves to the block Hessenberg setting.

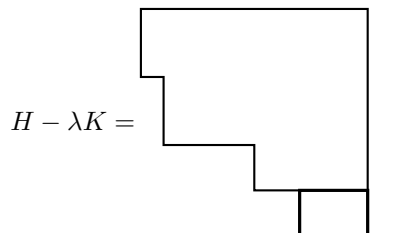
In this section, we will prove that the level- ρ poles will move down, but are retained. For simplicity, we will restrict the analysis to a single eigenvalue λ and eigenvector, though the principle generalizes readily to deflating subspaces.

Suppose our pencil $H - \lambda K$ to be of the following block Hessenberg form:

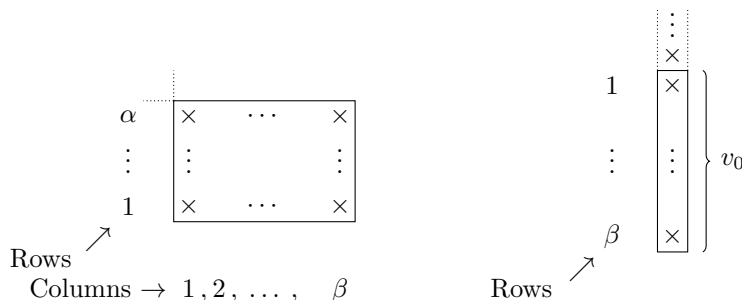


The proof is by following a procedure described in detail hereafter and proceeds by induction.

6.1. The initialization. The procedure starts, by acting on the marked block, in the lower-right corner:



For simplicity, we will only show this bottom block denoted by $\tilde{H} - \lambda\tilde{K}$. Suppose the block is of the following form (left) and part of the eigenvector v is shown on the right, we denote this part as v_0 .



The procedure starts with creating zeros in the vector v_0 . We do this with rotation $G_{\beta-1}^{(r)}$ that zeroes out $v_0(\beta)$. This is repeated until we obtain a unitary transformation composed of $\beta - 1$ rotations $G_1^{(r)} \cdots G_{\beta-1}^{(r)}$ such that $G_1^{(r)} \cdots G_{\beta-1}^{(r)} v_0 = \|v_0\|_2 e_1$. The Hermitian conjugate of these transformations needs to be applied on the right of the pencil as well. We end up with the equation

$$\begin{aligned} \left((\tilde{H} - \sigma\tilde{K}) \left(G_1^{(r)} \cdots G_{\beta-1}^{(r)} \right)^H \right) \left(G_1^{(r)} \cdots G_{\beta-1}^{(r)} \right) v_0 &= 0 \\ \left(\tilde{H}_{\beta-1} - \sigma\tilde{K}_{\beta-1} \right) e_1 &= 0. \end{aligned}$$

As a consequence, the first column of the transformed pencil $\tilde{H}_{\beta-1} - \sigma\tilde{K}_{\beta-1}$ captures the information on the shift σ . Pictorially we get the following, where the σ 's in the first column denote the information captured

in the matrices $\tilde{H}_{\beta-1}$ and $\tilde{K}_{\beta-1}$, i.e., it is a shorthand notation to indicate that $(\tilde{H}_{\beta-1} - \sigma\tilde{K}_{\beta-1})e_1 = 0$. This means that the first columns of $\tilde{H}_{\beta-1}$ and $\tilde{K}_{\beta-1}$ are a multiple of each other.

$$\begin{array}{|c|c|c|c|c|} \hline \sigma & \times & \cdots & \times & \\ \hline \vdots & \vdots & & \vdots & \\ \hline \sigma & \times & \cdots & \times & \\ \hline \end{array}
 \quad
 \left.
 \begin{array}{|c|} \hline \vdots \\ \vdots \\ \times \\ \times \\ 0 \\ \vdots \\ 0 \\ \hline \end{array}
 \right\} v_{\beta-1}$$

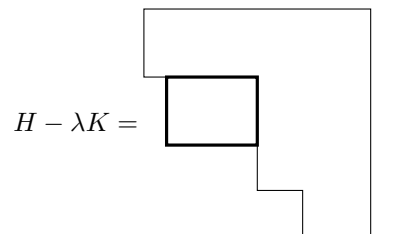
The next steps of the algorithm consist of transforming the first columns of $\tilde{H}_{\beta-1}$ and $\tilde{K}_{\beta-1}$ to a multiple of e_1 by applying rotations on the rows. As both columns are multiples of each other, we can do this simultaneously. Pictorially, after these transformations, we end up with:

$$\begin{array}{|c|c|c|c|c|} \hline \sigma & \times & \cdots & \times & \\ \hline 0 & \times & \cdots & \times & \\ \hline \vdots & \vdots & & \vdots & \\ \hline 0 & \times & \cdots & \times & \\ \hline \end{array}
 \quad
 \begin{array}{|c|} \hline \vdots \\ \vdots \\ \times \\ \times \\ 0 \\ \vdots \\ 0 \\ \hline \end{array}$$

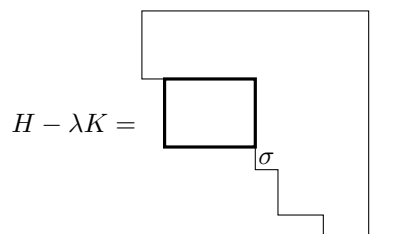
So we see that after the transformations, the block in the bottom-right corner of the pencil $H - \lambda K$ has moved down and to the right, and that it has lost a row and a column. The shift σ has taken up its spot in the top left corner, isolated from the other blocks. This concludes the initialization procedure, we can now go to the induction procedure to see that all level- ρ poles move down.

One can wonder what happened to the level- ρ poles that are/were present in the block in the lower-right corner. We come back to that after the next section.

6.2. The induction procedure. Let us focus now on a particular block in the block Hessenberg pencil. For example, consider the marked block, in the original, unaltered pencil $H - \lambda K$:

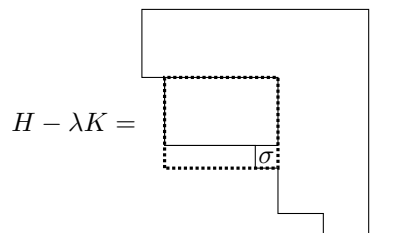


Starting from the lower-right block, the eigenvalue σ is moved to the left-upper position in that block. For the subsequent blocks, this eigenvalue is moved at the bottom of the block from right to left by rotations applied to the right of the pencil and then from bottom to top using rotations applied to the left of the pencil. After several steps of the algorithm, we arrive at the situation



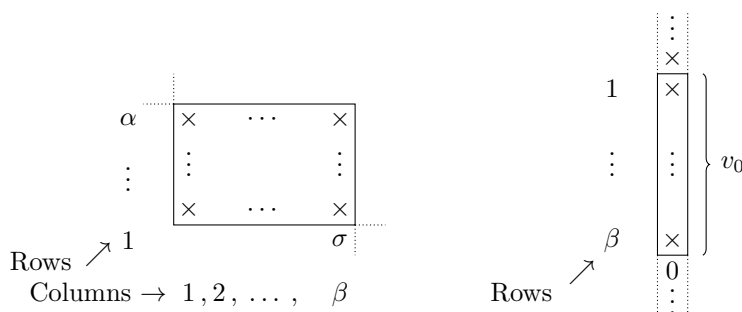
where the element σ means that σ is the eigenvalue of that particular 1 by 1 block. It is not the actual element of $H - \lambda K$.

From now we will focus only on the block marked with the dotted line below:



We will prove that the block moves down a single row and that all level- ρ poles are preserved.

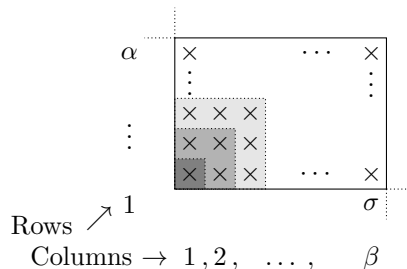
After thus having executed part of the algorithm we have arrived at the following situation where, in this particular example, the number of columns β in the block, is larger than the number of rows α , that is $\beta \geq \alpha$:



On the left, we see part of the matrix, that will be subjected to the action. The eigenvector, displayed on the right, has been partially transformed already to transform the vector to a multiple of e_1 . We denote, again as in the previous section, with v_0 the part where the action will take place. Every rotation that is applied on the rows of the vector will need to be applied from the right to act on the columns of the displayed matrix. We have clearly picked parts out of the entire matrix and vector that match when executing the transformations. We index the rows by $\alpha, \alpha - 1, \dots, 1$ from top to bottom and the columns by $1, 2, \dots, \beta$ from left to right. We denote the partially transformed matrix pencil as (H_0, K_0) and the perfect shift by σ .

Though we have taken a particular case here where $\beta \geq \alpha$, all other cases are similar. We will always end up with a block, possibly rectangular, and a single element. The single element will contain the eigenvalue one wishes to deflate. In the case of deflating subspaces, this will be a block, having the eigenvalues one wishes to deflate, but even then the procedure goes similarly.

Before proceeding with the flow of the algorithm and studying how the level- ρ poles will move down, we will identify the poles in this example. Note that the position of the diagonal in the full matrix is not specified, in fact it is not essential in the analysis to know the exact position.



Here, the element marked in \blacksquare denotes a level ρ -pole (eigenvalue of $H_0(1, 1) - \lambda K_0(1, 1)$); the eigenvalues of the block marked in \blacksquare are level $(\rho - 1)$ -poles (eigenvalues of $H_0(1 : 2, 1 : 2) - \lambda H_0(1 : 2, 1 : 2)$); and the eigenvalues of the 3×3 block marked as \blacksquare are level $(\rho - 2)$ -poles (eigenvalues of $H_0(1 : 3, 1 : 3) - \lambda H_0(1 : 3, 1 : 3)$). In the forthcoming analysis, we will prove that these poles will be preserved.

We assume $v_0(\beta) \neq 0$. When $v_0(\beta) = 0$, the rotations applied at this point can be taken as the identity transformations not changing the bottom-right part of the pencil. Hence, the level- ρ poles are trivially maintained. In case $v_0(\beta) \neq 0$

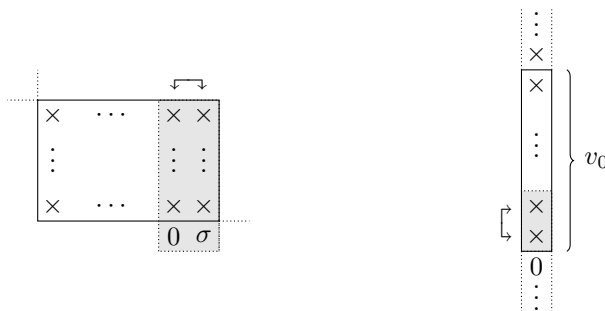
$$(H_0(1, \beta) - \sigma K_0(1, \beta))v_0(\beta) = 0,$$

or

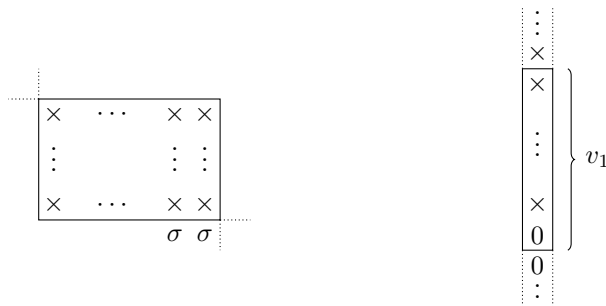
$$H_0(1, \beta) - \sigma K_0(1, \beta) = 0.$$

So, σ is an eigenvalue of the 1×1 pencil $(H_0(1, \beta), K_0(1, \beta))$.

The next part of the algorithm consists of $(\beta - 1)$ steps. In each step, the shift σ is moved one place to the left. In step 1, $v_0(\beta)$ is made zero by applying a rotation $G_{\beta-1}^{(r)}$ to the vector $[v_0(\beta - 1), v_0(\beta)]^T$. To maintain the equality $(H_0 - \sigma K_0)v_0 = 0$, we multiply H_0 and K_0 to the right by $(G_{\beta-1}^{(r)})^H$. As a result, we end up with $\left((H_0 - \sigma K_0) \left(G_{\beta-1}^{(r)} \right)^H \right) G_{\beta-1}^{(r)} v_0 = 0$. Graphically, we arrive at the following figure, where the rotation is displayed acting on both the matrix and the vector, affecting the parts marked in lightgray.



The transformed matrix pencil (H_1, K_1) and the eigenvector have the following form:



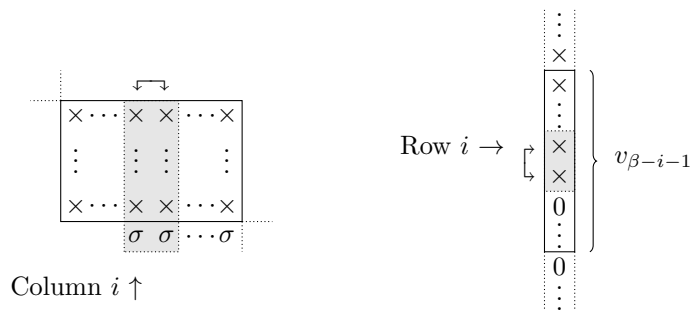
Since $(H_1 - \sigma K_1) G_{\beta-1}^{(r)} v_0 = 0$, we can argue that the shift σ is an eigenvalue of the matrix pencil $(H_1(1, \beta - 1), K_1(1, \beta - 1))$. Also in the lower-right corner the information on the shift σ is still present, we have marked this in the figure by letting the σ be there as well, whenever we put a σ in the figures this means that the information on the shift is in there.

Suppose we have already performed $(\beta - i - 1)$ steps. In step $(\beta - i - 1)$, we make $v_{\beta-i-1}(i + 1)$ zero by applying a rotation $G_i^{(r)}$ to the vector $[v_{\beta-i-1}(i), v_{\beta-i-1}(i + 1)]^T$. To maintain the equality

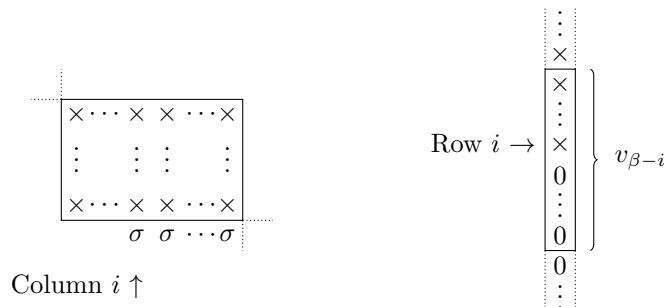
$$(H_{\beta-i-1} - \sigma K_{\beta-i-1})v_{\beta-i-1} = 0,$$

we multiply $H_{\beta-i-1}$ and $K_{\beta-i-1}$ to the right by $(G_i^{(r)})^H$, that is, we get

$$\left((H_{\beta-i-1} - \sigma K_{\beta-i-1})(G_i^{(r)})^H \right) G_i^{(r)} v_{\beta-i-1} = 0.$$



The transformed matrix pencil $(H_{\beta-i}, K_{\beta-i})$ and the eigenvector $v_{\beta-i}$ have the following form:



And again, the pencil $(H_{\beta-i}(1, i), K_{\beta-i}(1, i))$ has eigenvalue σ .

After performing step $\beta - 1$, the transformed pencil $(H_{\beta-1}, K_{\beta-1})$ and the eigenvector $v_{\beta-1}$ have the following form:

$$\begin{array}{c}
 \begin{array}{|cccc|}
 \hline
 \sigma & \times & \cdots & \times \\
 \vdots & \vdots & & \vdots \\
 \sigma & \times & \cdots & \times \\
 \hline
 \sigma & \sigma & \cdots & \sigma \\
 \hline
 \end{array}
 &
 \begin{array}{c}
 \vdots \\
 \times \\
 \times \\
 0 \\
 \vdots \\
 \vdots \\
 0 \\
 \vdots \\
 \vdots
 \end{array}
 \end{array}
 \left. \vphantom{\begin{array}{c} \vdots \\ \times \\ \times \\ 0 \\ \vdots \\ \vdots \\ 0 \\ \vdots \\ \vdots \end{array}} \right\} v_{\beta-1}$$

The shift σ appears not only in the last row but also in the first column, so in positions $(1 : \alpha, 1)$ of the transformed matrix pencil.

In the next $(\alpha - 1)$ steps, the elements $(1, 1), (2, 1), \dots, (\alpha - 1, 1)$ of the matrices $H_{\beta-1}$ and $K_{\beta-1}$ are made equal to zero by multiplications to the left by rotations; as both columns are multiples of each other this is possible to do this simultaneously. So, in the first step, element $(1, 1)$ is made equal to zero. The next figure shows the rows that will be affected, and it also shows the rotation that will be applied.

$$\begin{array}{c}
 \begin{array}{|cccc|}
 \hline
 \sigma & \times & \cdots & \times \\
 \vdots & \vdots & & \vdots \\
 \sigma & \times & \cdots & \times \\
 \hline
 \sigma & \sigma & \cdots & \sigma \\
 \hline
 \end{array}
 &
 \begin{array}{c}
 \vdots \\
 \times \\
 \times \\
 0 \\
 \vdots \\
 \vdots \\
 0 \\
 \vdots \\
 \vdots
 \end{array}
 \end{array}
 \left. \vphantom{\begin{array}{c} \vdots \\ \times \\ \times \\ 0 \\ \vdots \\ \vdots \\ 0 \\ \vdots \\ \vdots \end{array}} \right\} v_{\beta-1}$$

Hence, the transformed pencil (H_{β}, K_{β}) and the eigenvector $v_{\beta} = v_{\beta-1}$ look as follows:

$$\begin{array}{c}
 \begin{array}{|cccc|}
 \hline
 \sigma & \times & \cdots & \times \\
 \vdots & \vdots & & \vdots \\
 \sigma & \times & \cdots & \times \\
 \hline
 0 & \times & \cdots & \times \\
 \hline
 \end{array}
 &
 \begin{array}{c}
 \vdots \\
 \times \\
 \times \\
 0 \\
 \vdots \\
 \vdots \\
 0 \\
 \vdots \\
 \vdots
 \end{array}
 \end{array}
 \left. \vphantom{\begin{array}{c} \vdots \\ \times \\ \times \\ 0 \\ \vdots \\ \vdots \\ 0 \\ \vdots \\ \vdots \end{array}} \right\} v_{\beta} = v_{\beta-1}$$

Looking at the eigenvalues of the pencil $(H_{\beta}(1 : 2, 1 : 2), K_{\beta}(1 : 2, 1 : 2))$ and of the pencil $(H_{\beta-2}(1 : 2, 1 : 2), K_{\beta-2}(1 : 2, 1 : 2))$ and the rotations applied, we see that these eigenvalues are the same. The pencils

are upper triangular 2×2 . Because the shift swaps on the diagonal, also the other eigenvalue is maintained and swapped. This way of reasoning is continued.

In step 2, element $(2,1)$ is made equal to zero by applying a rotation to the left, leading to the pencil $(H_{\beta+1}, K_{\beta+1})$ and $v_{\beta+1} = v_{\beta-1}$.

$$\begin{array}{|c|} \hline \vdots \\ \hline \times \\ \hline \times \\ \hline 0 \\ \hline \vdots \\ \hline 0 \\ \hline \vdots \\ \hline \end{array} \left. \vphantom{\begin{array}{|c|} \hline \vdots \\ \hline \times \\ \hline \times \\ \hline 0 \\ \hline \vdots \\ \hline 0 \\ \hline \vdots \\ \hline \end{array}} \right\} v_{\beta+1} = v_{\beta} = v_{\beta-1}$$

$$\begin{array}{|c|} \hline \sigma \times \cdots \times \\ \hline \vdots \vdots \vdots \\ \hline \sigma \times \cdots \times \\ \hline 0 \times \cdots \times \\ \hline 0 \times \cdots \times \\ \hline \end{array}$$

Looking at the eigenvalues of the pencil shown on the left below, namely $(H_{\beta+1}(1 : 3, 1 : 3), K_{\beta+1}(1 : 3, 1 : 3))$ and of the pencil shown on the right, namely $(H_{\beta-3}(1 : 3, 1 : 3), K_{\beta-3}(1 : 3, 1 : 3))$ and the rotations applied, we can conclude that these eigenvalues are the same. The pencils are block upper triangular:

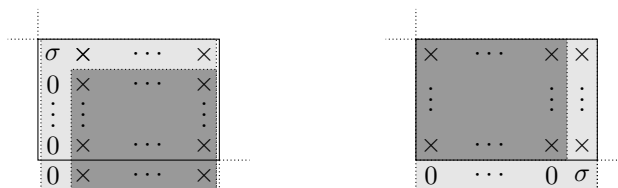
$$\begin{array}{|c|} \hline \sigma \times \times \cdots \times \\ \hline \vdots \vdots \vdots \\ \hline \sigma \times \times \cdots \times \\ \hline 0 \times \times \cdots \times \\ \hline 0 \times \times \cdots \times \\ \hline \end{array} \qquad \begin{array}{|c|} \hline \times \times \times \cdots \times \\ \hline \vdots \vdots \vdots \\ \hline \times \times \times \cdots \times \\ \hline \times \times \times \cdots \times \\ \hline 0 \ 0 \ \sigma \cdots \sigma \\ \hline \end{array}$$

After step $\alpha - 1$, we get the matrix pencil $(H_{\alpha+\beta-2}, K_{\alpha+\beta-2})$ and $v_{\alpha+\beta-2} = v_{\beta-1}$ of the following form:

$$\begin{array}{|c|} \hline \vdots \\ \hline \times \\ \hline \times \\ \hline 0 \\ \hline \vdots \\ \hline 0 \\ \hline \vdots \\ \hline \end{array} \left. \vphantom{\begin{array}{|c|} \hline \vdots \\ \hline \times \\ \hline \times \\ \hline 0 \\ \hline \vdots \\ \hline 0 \\ \hline \vdots \\ \hline \end{array}} \right\} v_{\alpha+\beta-2} = v_{\beta-1}$$

$$\begin{array}{|c|} \hline \sigma \times \cdots \times \\ \hline 0 \times \cdots \times \\ \hline \vdots \vdots \vdots \\ \hline 0 \times \cdots \times \\ \hline 0 \times \cdots \times \\ \hline \end{array}$$

The eigenvalues of the pencil $(H_{\alpha+\beta-2}(1 : \alpha, 1 : \alpha), K_{\alpha+\beta-2}(1 : \alpha, 1 : \alpha))$ (shown on the left below) and $(H_{\beta-\alpha}(1 : \alpha, 1 : \alpha), K_{\beta-\alpha}(1 : \alpha, 1 : \alpha))$ (shown on the right below) are the same. These pencils are block upper triangular:



Note also that the block shifted one place down along the diagonal as we have proven before.

Summarizing, the eigenvalues of the pencil

$$(H_{i+\beta-2}(2 : i, 2 : i), K_{i+\beta-2}(2 : i, 2 : i)) = (H(2 : i, 2 : i), K(2 : i, 2 : i)),$$

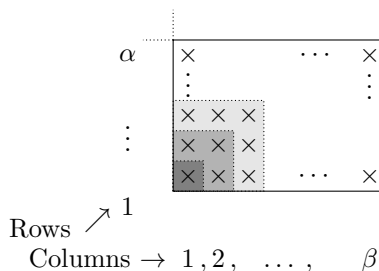
and

$$(H_{\beta-i}(1 : i - 1, 1 : i - 1), K_{\beta-i}(1 : i - 1, 1 : i - 1)) = (H_f(1 : i - 1, 1 : i - 1), K_f(1 : i - 1, 1 : i - 1)),$$

are the same for $i = 2, 3, \dots, \alpha$ where (H_f, K_f) denotes the final matrix pencil at the end of the algorithm.

6.3. Level- ρ poles in the block in the lower-right corner. After having initialized the procedure, we noted that the block in the lower-right corner moved down a row and moved a column to the right. The block has shrunk. As a consequence there is a bunch of level- ρ poles lost. More precisely, considering, for a particular ρ the eigenvalues of the pencil $H_{p\rho} - \lambda K_{p\rho}$ as in Definition 4.3, then, for the block in the lower-right corner, after the initialization, one eigenvalue will be gone.

Let us examine this in more detail. Consider the block in the lower-right corner of the following form:



After the initialization, the block will have lost one column and one row, as a consequence the element marked in \blacksquare , which denotes a level ρ -pole (eigenvalue of $H_0(1, 1) - \lambda K_0(1, 1)$) will have been lost. For the block marked in \blacksquare , which are the level $(\rho - 1)$ -poles (eigenvalues of $H_0(1 : 2, 1 : 2) - \lambda H_0(1 : 2, 1 : 2)$) one eigenvalue will disappear. The same holds for the eigenvalues of the 3×3 block marked as \blacksquare , which are the level $(\rho - 2)$ -poles (eigenvalues of $H_0(1 : 3, 1 : 3) - \lambda H_0(1 : 3, 1 : 3)$). Let us only examine the level $(\rho - 1)$ -poles.

To prove this, we need, however, a little preprocessing. We compute the Schur decomposition of the block marked in \blacksquare and execute the corresponding unitary equivalence transformation to end up with the following situation, where also the vector v_0 has been transformed into \hat{v}_0 :

$$\begin{array}{|c|c|c|c|}
 \hline
 & & & \vdots \\
 \hline
 & \times & \times & \cdots & \times \\
 \hline
 & \vdots & \vdots & & \vdots \\
 \hline
 & \times & \times & \cdots & \times \\
 \hline
 & 0 & \times & \cdots & \times \\
 \hline
 \end{array}
 \quad
 \begin{array}{|c|}
 \hline
 \vdots \\
 \hline
 \times \\
 \hline
 \times \\
 \hline
 \vdots \\
 \hline
 \times \\
 \hline
 \end{array}
 \left. \vphantom{\begin{array}{|c|} \hline \vdots \\ \hline \times \\ \hline \times \\ \hline \vdots \\ \hline \times \\ \hline \end{array}} \right\} \hat{v}_0$$

Both level $(\rho - 1)$ -poles can be identified easily now, and in the end, this preprocessing transformation will not have any effect at all on the final result, it is only an intermediate transformation that will allow us to identify what will happen. We will see now that only one of these two level $(\rho - 1)$ -poles will be retained, namely the one in the first column, the one in the last row will be lost.

After having executed $\beta - 2$ steps in transforming \hat{v}_0 to e_1 , we end up with the following situation:

$$\begin{array}{|c|c|c|c|}
 \hline
 & & & \vdots \\
 \hline
 & \times & \times & \cdots & \times \\
 \hline
 & \vdots & \vdots & & \vdots \\
 \hline
 & \times & \times & \cdots & \times \\
 \hline
 & 0 & \sigma & \cdots & \times \\
 \hline
 \end{array}
 \quad
 \begin{array}{|c|}
 \hline
 \vdots \\
 \hline
 \times \\
 \hline
 \times \\
 \hline
 0 \\
 \hline
 \vdots \\
 \hline
 0 \\
 \hline
 \end{array}
 \left. \vphantom{\begin{array}{|c|} \hline \vdots \\ \hline \times \\ \hline \times \\ \hline 0 \\ \hline \vdots \\ \hline 0 \\ \hline \end{array}} \right\} \hat{v}_{\beta-2}$$

It is important to note that the last row only contains one element equal to σ (in fact σ has overwritten one level $(\rho - 1)$ -pole). This is, however, enough to reconsider the analysis executed in the previous section (the induction procedure) to see that σ and the other level $(\rho - 1)$ -pole will in the end have swapped position.

The same analysis can be made for all other level ρ -poles as well to illustrate that they all will decrease in size, and several poles will get lost.

7. Numerical results. In this section, we report some numerical tests. All the computations were performed with `Matlab ver. R2022b` with machine precision $\epsilon_M \approx 2.22 \times 10^{-16}$.

A Hessenberg-2 ($H2$) matrix is a matrix with zeros below the second subdiagonal. We consider 10000 Hessenberg-2 – Hessenberg-2 ($H2H2$) matrix pencils $(H^{(i)}, K^{(i)})$, of size 100, with pseudo-random values drawn from the standard normal distribution (generated by the function `randn` of `Matlab`) as entries, and scaled such that $\|H^{(i)}\|_2 = \|K^{(i)}\|_2 = 1$. For each pencil, we randomly pick a real and a complex conjugate eigenpair $(\lambda^{(i)}, x^{(i)})$, computed by the classical QZ algorithm, and apply the perfect shift technique to deflate that particular eigenvalue from the matrix pencil obtaining the new $H2H2$ matrix pencil $(\hat{H}^{(i)}, \hat{K}^{(i)})$. Furthermore, we also apply the improved scaled residual approach, described in Section 5 of [7], to compute a better approximation $(\hat{\lambda}^{(i)}, \hat{x}^{(i)})$ of the eigenpair. This is achieved using the scale factors

$$d(1) = 1, \quad d(i + 1) = 2^{\text{round} \log_2 \|x^{(i:n)}\|_2}.$$

Then, $d(i + 1)/\sqrt{2} \leq \|x^{(i : n)}\|_2 \leq d(i + 1)\sqrt{2}$ and $d(i + 1) \leq d(i)$ since $\|x^{(i : n)}\|_2 \leq \|x^{(i - 1 : n)}\|_2$. One then applies one step of iterative refinement on the scaled eigenpair $(\hat{\lambda}^{(i)}, D^{-1}\hat{x}^{(i)})$, using the scaled pencil

$$H_d^{(i)} := D^{-1}H^{(i)}D, \quad K_d^{(i)} := D^{-1}K^{(i)}D, \quad \text{with } D := \text{diag}(d(1), \dots, d(n)),$$

and then transforms the resulting eigenpair back to the original scaling. It was shown in [7] that this scaling technique typically yields an improved eigenpair for the subsequent deflation, when using the perfect shift technique. This then yields the $H2H2$ matrix pencils $(\hat{H}^{(i)}, \hat{K}^{(i)})$. We define $b^{(i)} = 1/\sqrt{1 + |\lambda^{(i)}|^2}$, $a^{(i)} = \lambda^{(i)} b^{(i)}$, $\hat{b}^{(i)} = 1/\sqrt{1 + |\hat{\lambda}^{(i)}|^2}$, $\hat{a}^{(i)} = \hat{\lambda}^{(i)} \hat{b}^{(i)}$. The results are depicted in the histograms displayed in the following pictures. In each figure, the histogram to the left refers to the matrix pencils $(\tilde{H}^{(i)}, \tilde{K}^{(i)})$, while the one to the right refers to the $H2H2$ matrix pencils $(\hat{H}^{(i)}, \hat{K}^{(i)})$.

The first four figures concern the deflation of a real eigenpair, while the last three figures refer to the complex conjugate case.

Theoretically, the values of the transformed pencil should be zero in the first column under the diagonal and in the subsequent columns under the second subdiagonal. In finite precision, we look at the values

$$(7.5) \quad \log_{10} \sqrt{\|\text{tril}(\tilde{H}^{(i)}, -3)\|_F^2 + \|\text{tril}(\tilde{K}^{(i)}, -3)\|_F^2 + s^2},$$

with

$$s^2 = |\tilde{H}_{2,1}^{(i)}|^2 + |\tilde{K}_{2,1}^{(i)}|^2 + |\tilde{H}_{3,1}^{(i)}|^2 + |\tilde{K}_{3,1}^{(i)}|^2.$$

Here $\text{tril}(M, k)$ denotes the matrix consisting of the diagonals below and on subdiagonal k . In Fig. 1, to the left the histogram of these values is displayed and to the right those of the pencil $(\hat{H}^{(i)}, \hat{K}^{(i)})$. It can be noticed that if the improved scaled residual approach is not applied, the part below the second subdiagonal of the computed $H2H2$ matrices often gets blurred.

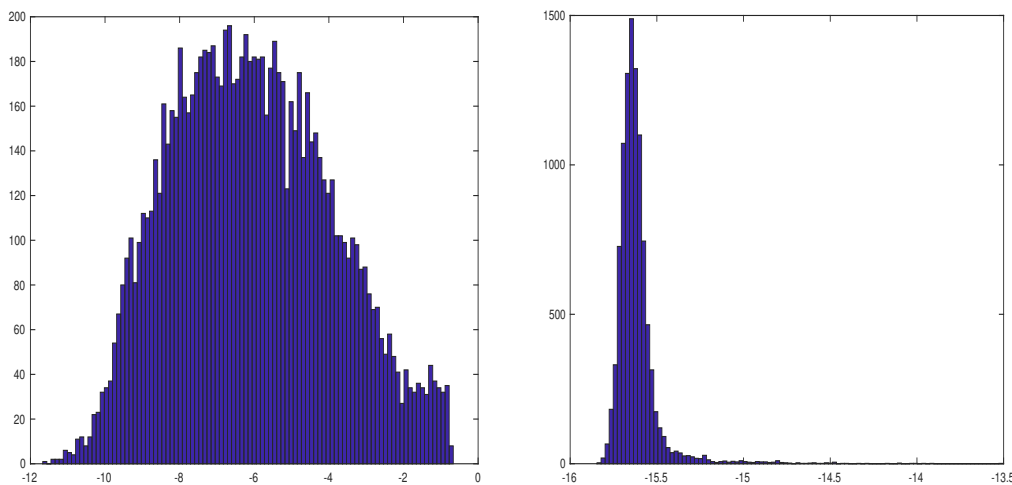


FIGURE 1. Histograms of the values (7.5) for $(\tilde{H}^{(i)}, \tilde{K}^{(i)})$ (left) and for $(\hat{H}^{(i)}, \hat{K}^{(i)})$ (right), real eigenvalue.

In Fig. 2, the histograms of \log_{10} of the chordal distance between (a_0, b_0) and $(\tilde{H}_{1,1}, \tilde{K}_{1,1})$ (left) and $(\hat{H}_{1,1}, \hat{K}_{1,1})$ (right) are displayed.

In Fig. 3, the histograms of the logarithms of the residuals $\log_{10} \|(aK - bH)x\|_2$ (left) and $\log_{10} \|(\hat{a}K - \hat{b}H)\hat{x}\|_2$ (right) are displayed.

The histograms in Fig. 4 show the accuracy of the poles in the $H2H2$ matrices after deflation. In particular, defined by $p_j = H_{j+2,j}/K_{j+2,j}$, $\tilde{p}_j = \tilde{H}_{j+3,j+1}/\tilde{K}_{j+3,j+1}$, and $\hat{p}_j = \hat{H}_{j+3,j+1}/\hat{K}_{j+3,j+1}$, $j = 1, \dots, n - 3$, the values of $\log_{10} \max_j \frac{|p_j - \tilde{p}_j|}{|p_j|}$ (left) and $\log_{10} \max_j \frac{|p_j - \hat{p}_j|}{|p_j|}$ (right) are displayed.

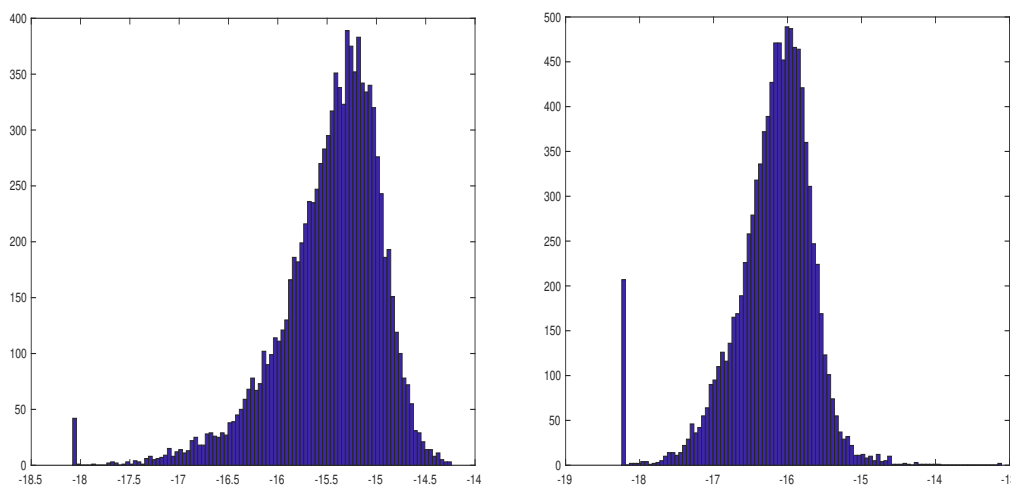


FIGURE 2. Histograms of \log_{10} of the chordal distance between (a_0, b_0) and $(\tilde{H}_{1,1}, \tilde{K}_{1,1})$ (left) and $(\hat{H}_{1,1}, \hat{K}_{1,1})$ (right), real eigenvalue.

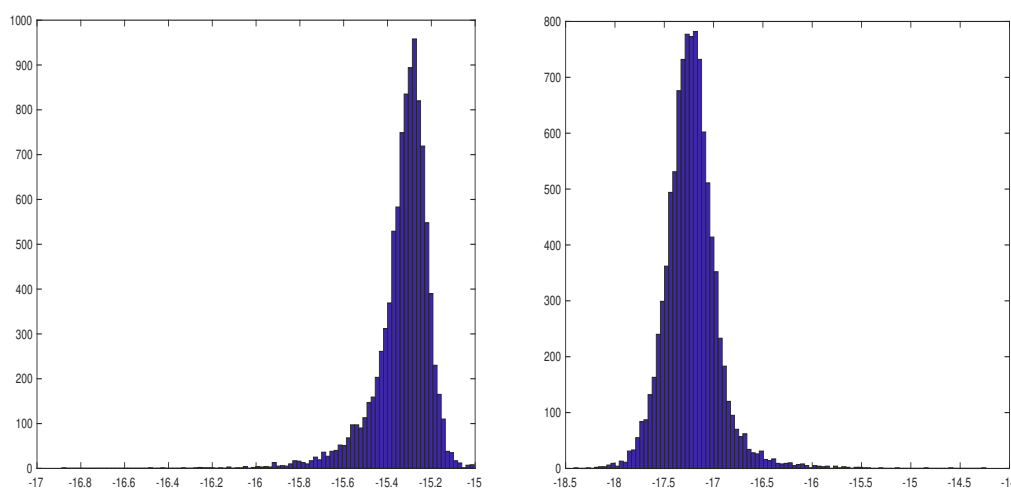


FIGURE 3. Histograms of the logarithms of the residuals, $\log_{10} \|(aK - bH)x\|_2$ (left) and $\log_{10} \|(\hat{a}K - \hat{b}H)\hat{x}\|_2$ (right), real eigenvalue.

The next three figures report the histograms corresponding to the complex conjugate eigenpair $(\lambda^{(i)}, \overline{\lambda^{(i)}})$. In Fig. 5, to the left the histogram of the values

$$(7.6) \quad \log_{10} \sqrt{\|\text{tril}(\tilde{H}^{(i)}, -3)\|_F^2 + \|\text{tril}(\tilde{K}^{(i)}, -3)\|_F^2 + s^2}$$

with

$$s^2 = |\tilde{H}_{3,1}^{(i)}|^2 + |\tilde{K}_{3,1}^{(i)}|^2 + |\tilde{H}_{3,2}^{(i)}|^2 + |\tilde{K}_{3,2}^{(i)}|^2 + |\tilde{H}_{4,2}^{(i)}|^2 + |\tilde{K}_{4,2}^{(i)}|^2.$$

is displayed and to the right those of the pencil $(\hat{H}^{(i)}, \hat{K}^{(i)})$. Similarly to the real case, it can be noticed that if the improved scaled residual approach is not applied, the part below the second subdiagonal of the computed $H2H2$ matrices get blurred.

In Fig. 6, the histograms of the logarithms of the residuals $\log_{10} \|(aK - bH)x\|_2$ (left) and $\log_{10} \|(\hat{a}K - \hat{b}H)\hat{x}\|_2$

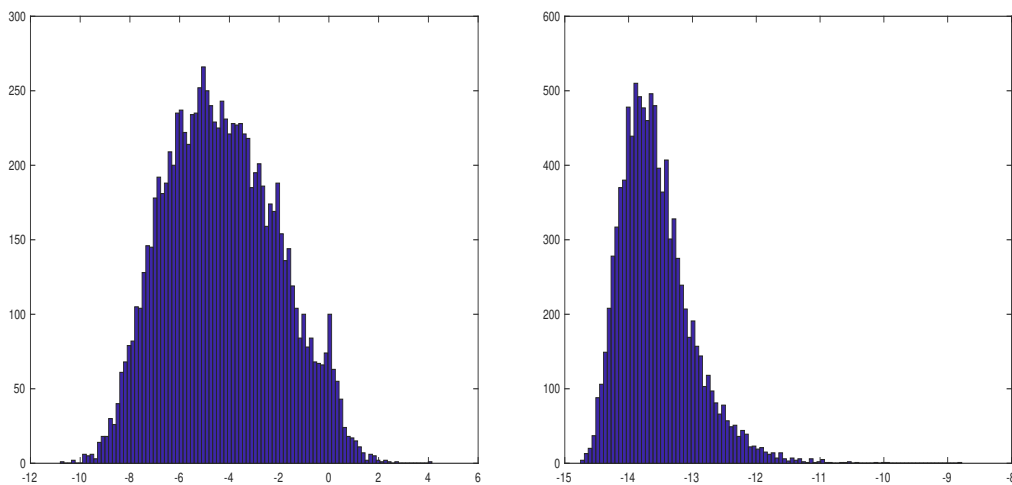


FIGURE 4. Accuracy of the poles in the $H2H2$ matrices after deflation, real eigenvalue.

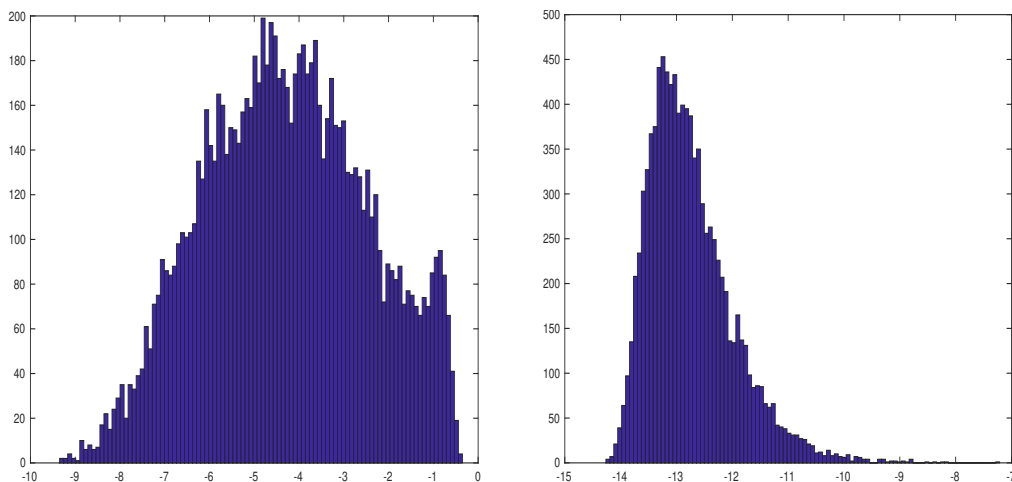


FIGURE 5. Histograms of the values (7.6) for $(\tilde{H}^{(i)}, \tilde{K}^{(i)})$ (left) and for $(\hat{H}^{(i)}, \hat{K}^{(i)})$ (right), complex conjugate eigenpair.

(right), are displayed.

The histograms in Fig. 7 display the accuracy of the poles in the $H2H2$ matrices after deflation. In particular, defined $p_j = H_{j+2,j}/K_{j+2,j}$, $\tilde{p}_j = \tilde{H}_{j+4,j+2}/\tilde{K}_{j+4,j+2}$, and $\hat{p}_j = \hat{H}_{j+4,j+2}/\hat{K}_{j+4,j+2}$, $j = 1, \dots, n - 4$, the values of $\log_{10} \max_j \frac{|p_j - \tilde{p}_j|}{|p_j|}$ (left) and $\log_{10} \max_j \frac{|p_j - \hat{p}_j|}{|p_j|}$ (right), are displayed.

In the next experiment, we consider 10000 HH pencils of size 8×8 with a bulge in position $(3 : 5, 2 : 4)$ using the improved scaled residual approach. In Figs. 8 and 9, we show the histograms of the maximum relative error between the eigenvalues of $(H_{5,2}, K_{5,2})$ and $(\hat{H}_{6,3}, \hat{K}_{6,3})$, of $(H_{4:5,2:3}, K_{4:5,2:3})$ and $(\hat{H}_{5:6,3:4}, \hat{K}_{5:6,3:4})$, of $(H_{3:5,2:4}, K_{3:5,2:4})$ and $(\hat{H}_{4:6,3:5}, \hat{K}_{4:6,3:5})$, respectively.

As a last example, we consider 10000 Hessenberg-like, Hessenberg pencils of size 100×100 using the improved scaled residual approach. All matrices below or on the diagonal of H have rank 1. This should also be true for \hat{H} . In Fig. 10 on the left, the histogram is shown of the 2-norm of all singular values of

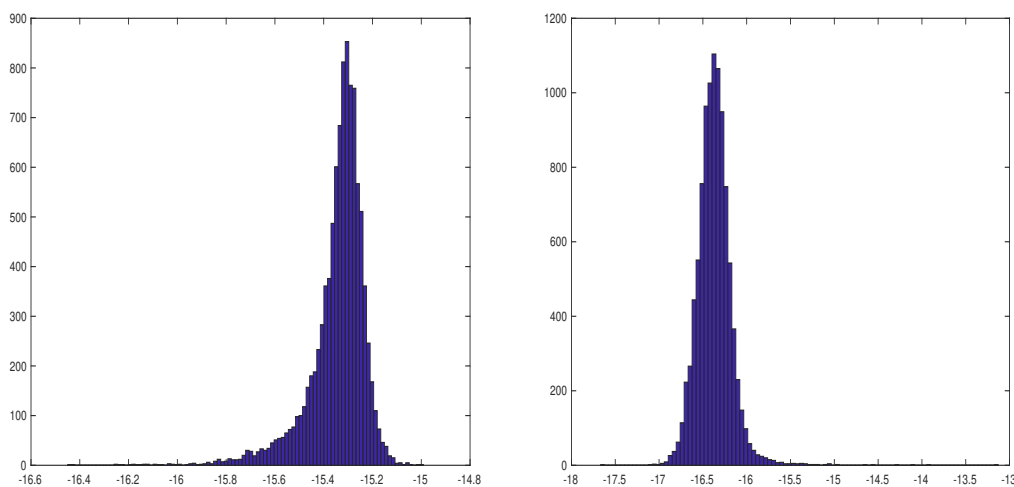


FIGURE 6. Histograms of the logarithms of the residuals $\log_{10} \|(aK - bH)x\|_2$ (left) and $\log_{10} \|(\hat{a}K - \hat{b}H)\hat{x}\|_2$ (right), complex conjugate eigenpair.

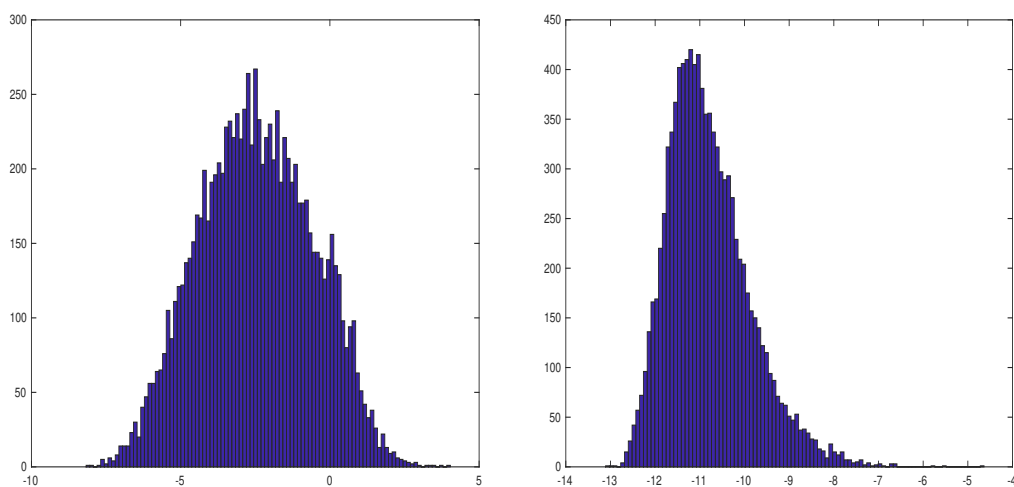


FIGURE 7. Accuracy of the poles in the HH matrices after deflation, complex conjugate eigenpair.

the matrices $\hat{H}_{j:n,1:j}$ for $j = 1, 2, \dots, n$ except for the highest singular value. In Fig. 10 on the right, the histogram is displayed of the values

$$(7.7) \quad \log_{10} \sqrt{\|\text{tril}(\tilde{K}^{(i)}, -2)\|_F^2 + |\tilde{K}_{2,1}^{(i)}|^2}.$$

8. Conclusions. An algorithm to perform perfectly shifted QR steps on general rank structured pencils $H - \lambda K$ was presented. It was shown that the rank structure of the individual matrices H and K moves down along the diagonal. We have defined the level- ρ poles and have proven that for block Hessenberg pencils, the level- ρ poles are preserved. This observation could be the basis to enhance the numerical stability of corresponding deflating algorithms. The method presented does not only work for a single eigenvalue but can deal with deflating subspaces as well. Numerical experiments illustrate the viability of the presented approach.

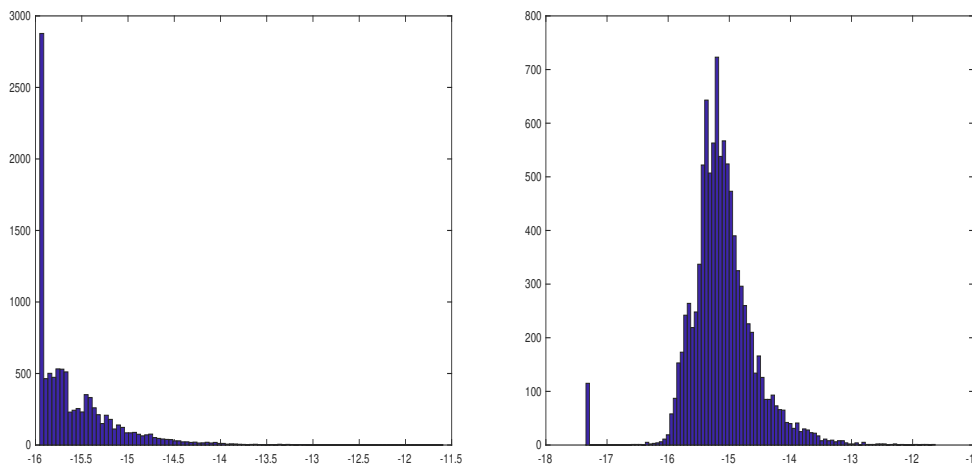


FIGURE 8. Maximum relative error between the eigenvalues of $(H_{5,2}, K_{5,2})$ and $(\hat{H}_{6,3}, \hat{K}_{6,3})$ (left) and of $(H_{4:5,2:3}, K_{4:5,2:3})$ and $(\hat{H}_{5:6,3:4}, \hat{K}_{5:6,3:4})$ (right).

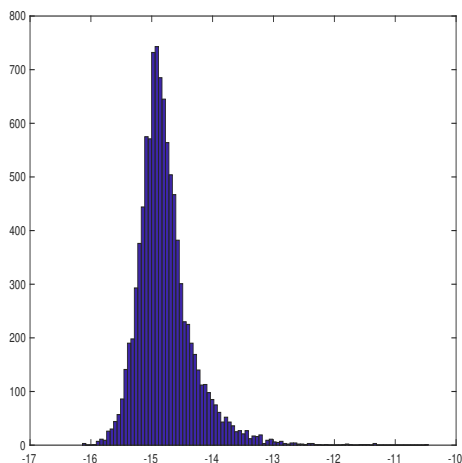


FIGURE 9. Maximum relative error between the eigenvalues of $(H_{3:5,2:4}, K_{3:5,2:4})$ and $(\hat{H}_{4:6,3:5}, \hat{K}_{4:6,3:5})$.

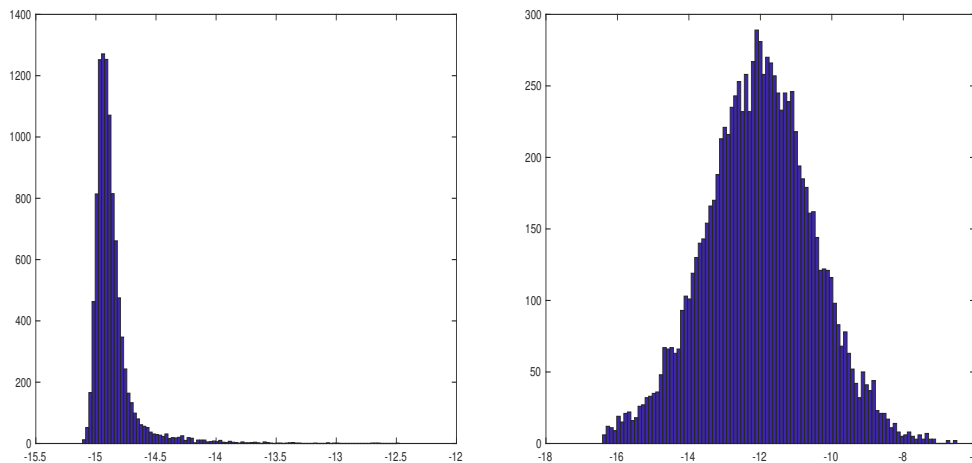


FIGURE 10. The 2-norm of all singular values of the matrices $\hat{H}_{j:n,1:j}$ for $j = 1, 2, \dots, n$ except for the highest singular value (left); the values (7.7) (right).

REFERENCES

- [1] J.L. Aurentz, T. Mach, L. Robol, R. Vandebril, and D.S. Watkins. Core-chasing algorithms for the eigenvalue problem. *Fund. Algor.*, 13, SIAM, 2018.
- [2] K. Braman, R. Byers, and R. Mathias. The multishift QR algorithm. Part I. Maintaining well-focused shifts and level 3 performance. *SIAM J. Matrix Anal. Appl.*, 23(4):929–947, 2002.
- [3] J. Dongarra and F. Sullivan. Introduction to the top 10 algorithms. *Comput. Sci. Eng.*, 2(1):22–23, 2000.
- [4] Y. Eidelman, I.C. Gohberg, and I. Haimovici. Separable type representations of matrices and fast algorithms – volume 2: Eigenvalue method. In: *Operator Theory: Advances and Applications*, vol. 235, Springer Basel, 2013.
- [5] N. Guglielmi, M. Overton, and G. Stewart. An efficient algorithm for computing the generalized null space decomposition. *SIAM J. Matrix Anal. Appl.*, 36:38–54, 2015.
- [6] N. Mastronardi and P. Van Dooren. On QZ steps with perfect shifts and computing the index of a differential-algebraic equation. *IMA J. Numer. Anal.*, 2020. doi: 10.1093/imanum/draa049.
- [7] N. Mastronardi, M. Van Barel, R. Vandebril, and P. Van Dooren, Rational QZ steps with perfect shifts. *Numer. Algor.*, 2023. doi: 10.1007/s11075-023-01600-2.
- [8] N. Mastronardi and P. Van Dooren. The QR steps with perfect shifts. *SIAM J. Matrix Anal. Appl.*, 39(4):1591–1615, 2018.
- [9] M. Myllykoski. Algorithm 1019: A task-based multi-shift QR/QZ algorithm with aggressive early deflation. *ACM Trans. Math. Softw.*, 48(1), Article11, 36 pages, 2021.
- [10] G.W. Stewart. Error and perturbation bounds for subspaces associated with certain eigenvalue problems. *SIAM Rev.*, 15:727–764, 1973.
- [11] R. Vandebril, M. Van Barel, and N. Mastronardi. *Matrix Computations and Semiseparable Matrices, Volume I: Linear Systems*. Johns Hopkins University Press, Baltimore, Maryland, USA, 2008.
- [12] R. Vandebril, M. Van Barel, and N. Mastronardi. *Matrix Computations and Semiseparable Matrices, Volume II: Eigenvalue and Singular Value Methods*. Johns Hopkins University Press, Baltimore, Maryland, USA, 2008.
- [13] R. Vandebril. Chasing bulges or rotations? A metamorphosis of the QR -algorithm. *SIAM J. Matrix Anal. Appl.*, 32: 217–247, 2011.
- [14] D.S. Watkins. The transmission of shifts and shift blurring in the QR algorithm. *Linear Algebra Appl.*, 241–243:877–896, 1996.