An implicit filter for rational Krylov using core transformations

Camps Daan[a,*], Meerbergen Karl[a], Vandebril Raf[a]

[a]*Department of Computer Science, KU Leuven - University of Leuven*
Celestijnenlaan 200A, 3001 Leuven, Belgium

Abstract

The rational Krylov method is a powerful tool for computing a selected subset of eigenvalues in large-scale eigenvalue problems. In this paper we study a method to implicitly apply a filter in a rational Krylov iteration by directly acting on a QR factorized representation of the Hessenberg pair from the rational Krylov method. This filter is used to restart the iteration, which is generally required to limit the orthogonalization and storage costs. The contribution in this paper is three-fold. We reformulate existing procedures in terms of operations on core transformations. This has the advantage of improved convergence monitoring. Secondly, we demonstrate that the extended QZ method is a special case of this more general method. Finally, numerical experiments show the validity and the increased accuracy of the new approach compared with existing methods.

Keywords: rational Krylov, extended Krylov, implicit, QZ, restart, filter, core transformations

AMS subject classifications. 15A18, 65F15

1. Introduction.

The Arnoldi algorithm, first introduced by Arnoldi (1951) [1] and studied extensively by Saad [2, 3, 4, 5] is the *classical* Krylov method. It is a projection method frequently used for solving systems of linear equations, eigenvalue problems, matrix equations, and so forth. The Arnoldi algorithm generates a subspace on which the problem is projected. The resulting smaller problem is then solved, and lifted back to its original dimensions.

The Arnoldi method has a particular convergence behavior. If we focus on eigenvalue computations, it locates first well-separated extreme eigenvalues [5, 6, 7]. Computing, e.g., rightmost eigenvalues, eigenvalues near the origin or eigenvalues in a certain region with prescribed accuracy may be infeasible with a small number of Krylov vectors. One way to reduce the number of vectors is by using methods that converge faster towards a particular region of interest. *Shift-and-invert* Arnoldi [8, 9, 10] uses the matrix $(A - \sigma I)^{-1}$ to compute the eigenvalues near the 'shift' σ. In the rational Krylov method (RK), introduced by Ruhe [11, 12, 13, 14, 15], the shift or *pole* may change at every step in the iteration. The extended Krylov method (EK), first proposed by Druskin & Knizhnerman (1998) [16] for the approximation of matrix functions, is a special case of the RK method that only uses shifts at zero and infinity.

In every step of the Arnoldi algorithm, an explicit orthogonalization of the new vector against all previously computed basis vectors is performed. Consequently, the computational cost and storage requirements increase as the algorithm progresses. This problem can be solved by a restart of the Arnoldi method. Sorensen (1992) introduced the implicitly restarted Arnoldi method (IRA) [17]. His algorithm applies implicitly shifted QR steps on the Arnoldi Hessenberg matrix. The IRA method was further analyzed by Morgan (1996) [18] and refined by Lehoucq & Sorensen (1996) [19]. Stewart (2001) [20] introduced the Krylov-Schur algorithm where a proper subspace is extracted from the Krylov subspace via the Schur decomposition. De Samblanx, Meerbergen

*Corresponding author
Email address: daan.camps@cs.kuleuven.be (Camps Daan)

& Bultheel (1997) [21] proposed an implicit restart method for rational Krylov methods. Their method uses an explicit QZ step on the RK Hessenberg pencil. Recently, Berljafa & Güttel [22] proposed a method to change the poles in the RK method and noticed that this procedure can be used to restart the RK method [22, Section 4.3]. In their paper no comparison is made with the explicit method of De Samblanx et al. [21].

The contribution of this paper is threefold. First, we reformulate the method of Berljafa & Güttel in terms of operations on a QR-factorized representation of the RK Hessenberg pencil. In this representation, the unitary matrix is stored as a sequence of *core transformations*. This extends the *core chasing* techniques, introduced by Vandebril [23] for the dense eigenvalue problem and further developed by Vandebril & Watkins [24, 25], beyond *condensed* matrices. As such it is effectively a reformulation of the rational QZ method [26] in terms of core transformations. This representation allows for an efficient storage scheme of the unitary matrices and admits an accurate deflation criterion [27]. Furthermore, we use this representation to study the structure of the projection on a rational Krylov subspace in Lemma 3.1. Our second contribution is to show how the special case of the extended Krylov method is linked to the more general rational Krylov setting in Section 5.1. The final contribution is a comparison between the implicit [22] and explicit [21] version of the filter step. We will show that the implicit filter step is faster, more accurate, and can outperform the method of De Samblanx et al.

The paper is organized as follows. In Section 2, we introduce the notion of Krylov subspaces and generalize this to rational Krylov subspaces. Section 3 introduces unitary *core transformations* and uses them to construct a factorized representation of the RK Hessenberg pair. In this section, we also study the structure of the projection on a rational Krylov subspace. We show how the core transformations and the poles of the RK subspace are linked and how the core transformations can be altered to change the poles in Section 4. The gist of this paper is presented in Section 5 which explains the implicit QZ step for restarting the RK iteration based on manipulating core transformations. Section 6 illustrates the algorithm with three numerical examples that indicate that the method can outperform comparable methods. Some concluding remarks are given in Section 7.

Matrices are denoted in upper-case letters (A), the element at row i and column j is denoted as $(A)_{i,j}$ or $a_{i,j}$, the entire column j as $\boldsymbol{a}_j$, the first j columns as A_j, vectors in boldface lower-case ($\boldsymbol{v}$), and matrices with one more row than columns are underlined ($\underline{H}$). MATLAB notation is used frequently for matrix indexing and slicing. $A^\dagger$ is the generalized inverse of A and $\mathcal{R}(A)$ is the range of A. The extended complex plane $\mathbb{C} \cup \{\infty\}$ is denoted as $\bar{\mathbb{C}}$. In $\bar{\mathbb{C}}$, we define $\alpha/0 = \infty$ for any nonzero $\alpha \in \mathbb{C}$.

2. Polynomial, extended and rational Krylov methods.

Let $A \in \mathbb{C}^{N\times N}$ be a matrix and $\boldsymbol{v} \in \mathbb{C}^N$ a nonzero vector. The $(m+1)$st order *polynomial Krylov subspace* generated by the matrix A with starting vector $\boldsymbol{v}$ is denoted by $\mathcal{K}_{m+1}(A, \boldsymbol{v})$ and defined as

$$\mathcal{K}_{m+1}(A, \boldsymbol{v}) = \text{span}\{\boldsymbol{v},\ A\boldsymbol{v},\ \ldots\ ,\ A^m\boldsymbol{v}\}. \tag{1}$$

Accurate approximations for some eigenvalues of the original matrix A can be extracted from $\mathcal{K}_{m+1}$, even for $m \ll N$. Here, $\mathcal{K}_{m+1}$ is shorthand notation for $\mathcal{K}_{m+1}(A, \boldsymbol{v})$. It is assumed throughout this article that all subspaces are of full dimension, therefore $\mathcal{K}_{m+1}$ is of dimension $m+1$. This implies that $\mathcal{K}_{m+1}$ is isomorphic to $\mathcal{P}_m$, the set of all polynomials of degree $\leq m$. This is the justification for the term polynomial Krylov subspace.

The *Krylov matrix* $[\boldsymbol{v},\ A\boldsymbol{v},\ \ldots\ ,\ A^m\boldsymbol{v}]$ is a matrix whose columns span $\mathcal{K}_{m+1}$. However, for all practical purposes, this representation is inadequate since the consecutive vectors will point increasingly in the direction of the dominant eigenvector. The columns of the Krylov matrix will soon become linearly dependent, causing any numerical algorithm to break down as the subspace is no longer expanded. The construction of an orthonormal basis of $\mathcal{K}_{m+1}$ is a standard procedure to overcome this difficulty. This leads to the Arnoldi algorithm when A is unsymmetric [1], or the

Lanczos three-term recurrence variant for Hermitian A [28]. For the remainder of this paper no specific structure is assumed for A, prompting us to Arnoldi's method. Throughout the Arnoldi algorithm, the Arnoldi recurrence relation,

$$A\,V_m = V_{m+1}\,\underline{H}_m, \tag{2}$$

holds. The columns of V_{m+1} form an orthonormal basis of $\mathcal{K}_{m+1}$ and $\underline{H}_m$ is an *unreduced*, upper Hessenberg matrix; unreduced means in this context that the subdiagonal entries of $\underline{H}_m$ are nonzero. The upper Hessenberg matrix consists of two parts: the orthogonalization coefficients of the basis vectors in the upper triangular part and the 2-norm of the basis vectors on the subdiagonal. Since the 2-norm of a vector is strictly larger than zero for any nonzero vector, $\underline{H}_m$ is unreduced unless $A\boldsymbol{v}_m \in \mathcal{K}_m$, a *lucky breakdown* of the Arnoldi method. We have already excluded this possibility by making the assumption that subspaces are of full dimension.

From the Arnoldi recurrence relation (2) and the orthonormality of V_{m+1} it follows that H_m, the $m \times m$ top square part of $\underline{H}_m$, is the orthogonal projection of A on $\mathcal{K}_m$:

$$H_m = V_m^*\,A\,V_m. \tag{3}$$

The eigenvalues ϑ_k of H_m are called *Ritz values* and $\boldsymbol{x}_k = V_m\,\boldsymbol{z}_k$ are the corresponding *Ritz vectors*, where $\boldsymbol{z}_k$ is the eigenvector of H_m associated with ϑ_k. The Ritz pairs satisfy a Galerkin condition: $(A\,\boldsymbol{x}_k - \vartheta_k\,\boldsymbol{x}_k) \perp \mathcal{K}_m$. Computing the Ritz values is the standard method to extract eigenvalue estimates from $\mathcal{K}_{m+1}$. They typically converge first to extreme eigenvalues that are well-separated from the rest of the spectrum of A [5, 6, 7].

Over the past few decades a lot of research efforts went into generalizations of this procedure with improved convergence on parts of the spectrum that are difficult to approximate with the Arnoldi method. These generalizations either adjust the extraction procedure or the search-space.

Rational Krylov (RK) subspaces [11, 12, 13, 15, 22, 29] are a generalization of the second kind. In an RK iteration, operators that are rational functions of A are used to expand the subspace. These are matrices of the form,

$$M = (\alpha A + \beta I)^{-1}(\gamma A + \delta I), \tag{4}$$

with $\alpha, \beta, \gamma, \delta$ parameters that can change in every iteration. M is a Möbius transformation of A with pole ξ at $-\beta/\alpha$ and zero ϱ at $-\delta/\gamma$. If $\alpha = 0$ the pole is said to be at ∞, likewise the zero is at ∞ if $\gamma = 0$. Changing the position of the zero ϱ does not alter the subspace, but changing the pole ξ does change the span of the subspace. It is clear that one should choose $\xi \neq \varrho$ as otherwise M reduces to the identity matrix and that ξ cannot coincide with an eigenvalue of A as otherwise M is undefined.

Let us formally define a rational Krylov subspace.

Definition 2.1. *Given a matrix $A \in \mathbb{C}^{N\times N}$, a nonzero vector $\boldsymbol{v} \in \mathbb{C}^N$ and a tuple of poles $\Xi = (\xi_1, \ldots, \xi_m)$ with every $\xi_i \in \bar{\mathbb{C}} \setminus \Lambda(A)$, the $(m+1)$st order rational Krylov subspace is defined as,*

$$\mathcal{K}^{rat}_{m+1}(A, \boldsymbol{v}, \Xi) = \mathcal{K}^{rat}_{m+1}(A, \boldsymbol{v}, q_m) \coloneqq q_m(A)^{-1}\mathcal{K}_{m+1}(A, \boldsymbol{v}), \tag{5}$$

with $q_m \in \mathcal{P}_m$, a polynomial with roots Ξ. Every root in Ξ equal to ∞ reduces the degree of q_m by one.

The rational Krylov or *rational Arnoldi* algorithm listed in Algorithm 1 is a method for iteratively computing an orthonormal basis of $\mathcal{K}^{\mathrm{rat}}_{m+1}(A, \boldsymbol{v}, \Xi)$.

In line 3 of Algorithm 1, the zero is chosen as well as the vector $\boldsymbol{t}_j$ which is known as the *continuation vector*. This vector is chosen in such a way that it ensures that the subspace is expanded. A common choice proposed by Ruhe [15] is:

$$\boldsymbol{t}_j = \begin{cases} \boldsymbol{e}_j & \text{if } \xi_j = \xi_{j-1} \\ \boldsymbol{q}_j = Q_j\boldsymbol{e}_j & \text{otherwise} \end{cases},$$

Algorithm 1 rational Krylov algorithm [11, 12, 15] RK

Input: $A \in \mathbb{C}^{N\times N}$, $\mathbf{0} \neq \boldsymbol{v} \in \mathbb{C}^N$ and poles $\Xi = ((\alpha_i, \beta_i))_{j=1}^m = (\xi_i)_{j=1}^m$
Output: V_{m+1}, $\underline{K}_m$ and $\underline{L}_m$

1: $V_1 \coloneqq \boldsymbol{v}/\|\boldsymbol{v}\|_2$
2: **for** $j = 1, 2, \ldots, m$ **do**
3: Choose $\varrho_j = (\gamma_j, \delta_j)$ and $\boldsymbol{t}_j \in \mathbb{C}^j \setminus \{0\}$
4: Compute $\boldsymbol{w} \coloneqq (\alpha_j A + \beta_j I)^{-1}(\gamma_j A + \delta_j I)\, V_j \boldsymbol{t}_j$
5: Orthogonalize $\boldsymbol{w} \coloneqq \boldsymbol{w} - V_j\, \boldsymbol{h}_j$ where $\boldsymbol{h}_j = V_j^*\, \boldsymbol{w}$
6: Normalize $h_{j+1,j} \coloneqq \|\boldsymbol{w}\|_2$
7: Store new basis vector $V(:, j+1) \coloneqq \boldsymbol{w}/h_{j+1,j}$
8: Update recurrence matrices
a: $\underline{H}(1 : j+1, j) \coloneqq \underline{\boldsymbol{h}}_j = [\boldsymbol{h}_j^T\ h_{j+1,j}]^T$
b: $\underline{T}(1 : j, j) \coloneqq \boldsymbol{t}_j$
c: $D_\alpha(j, j) \coloneqq \alpha_j$, $D_\beta(j, j) \coloneqq \beta_j$, $D_\gamma(j, j) \coloneqq \gamma_j$, $D_\delta(j, j) \coloneqq \delta_j$
9: **end for**
return V_{m+1}, $\underline{K}_m \coloneqq \underline{H}_m D_\alpha - \underline{T}_m D_\gamma$ and $\underline{L}_m \coloneqq -\underline{H}_m D_\beta + \underline{T} D_\delta$.

where Q_j is computed from the QR factorization of $\alpha_j \underline{L}_{j-1} + \beta_j \underline{K}_{j-1}$. In any case the continuation vector should have a nonzero component in the direction of $\boldsymbol{q}_j$ as otherwise the space is not expanded [22].

Throughout Algorithm 1, the RK recurrence [12, 15, 21],

$$A\, V_{m+1}\, \underline{K}_m = V_{m+1}\, \underline{L}_m, \tag{6}$$

holds. This matrix decomposition is a generalization of the Arnoldi relation (2) and consists out of matrices $V_{m+1} \in \mathbb{C}^{N\times(m+1)}$ and a pair of upper Hessenberg matrices $(\underline{L}_m, \underline{K}_m) \in \mathbb{C}^{(m+1)\times m}$ that we will refer to as the RK Hessenberg pair. The columns of V_{m+1} are orthonormal by construction of Algorithm 1 and they span the RK subspace $\mathcal{K}_m^{\text{rat}}(A, \boldsymbol{v}, \Xi)$. The ratio of the subdiagonal elements of the upper Hessenberg matrices $\ell_{i,i+1}/k_{i,i+1}$ are equal to the poles ξ_i. The Hessenberg matrix $\underline{H}_m$ containing the orthogonalization coefficients in Algorithm 1 is again unreduced but the Hessenberg matrices $(\underline{L}_m, \underline{K}_m)$ are not guaranteed to each being unreduced separately. If $\xi_i = 0$ then $\ell_{i+1,i} = 0$ and similarly if $\xi_i = \infty$ then $k_{i+1,i} = 0$. Together, as a matrix pair, they are unreduced, which means that for all subdiagonal entries $|\ell_{i+1,i}| + |k_{i+1,i}| > 0$. Both $\underline{L}_m$ and $\underline{K}_m$ are of maximal rank m [21, 22].

As Berljafa & Güttel [22] have shown, there is an *essentially unique* one-to-one correspondence between an RK decomposition (6) and a RK subspace (5). We will call $(V_{m+1}, \underline{K}_m, \underline{L}_m)$ an RK *triplet* that corresponds to the RK subspace $\mathcal{K}_{m+1}^{\text{rat}}(A, \boldsymbol{v}_1, \Xi = (\ell_{i+1,i}/k_{i+1,i}))$.

Rational Krylov subspaces with all of their poles located at either 0 or ∞ form a special instance called *extended* Krylov (EK) subspaces. An example is given by:

$$\mathcal{K}_{m+1}^{\text{rat}}(A, \boldsymbol{v}, \Xi_{\text{ext}}) = \text{span}\{A^{-n}\boldsymbol{v},\ \ldots\ , A^{-1}\boldsymbol{v},\ \boldsymbol{v},\ A\boldsymbol{v},\ \ldots\ , A^p\boldsymbol{v}\}, \tag{7}$$

where n is equal to the number of entries in Ξ_{ext} that are zero and p is the number of poles at infinity. The dimension of the subspace is equal to $m{+}1{=}n{+}p{+}1$. It is clear that Arnoldi, the polynomial Krylov method, is a specific case of the EK method with $p{=}m$.

The Hessenberg pair $(\underline{L}_m, \underline{K}_m)$ related to an EK subspace has the property that it is in *condensed* format. This means that exactly one of the subdiagonal elements in the matrix pair is equal to zero at every row. So, either $\ell_{i+1,i} = 0$ or $k_{i+1,i} = 0$ for $i \in \{1, \ldots, m\}$. Consequently the *condensed* QZ method of Vandebril & Watkins [25] can be applied to the EK Hessenberg pencil $(\underline{L}_m, \underline{K}_m)$ to filter the EK method. We will show in Section 5 that this is a special case of the RK filter.

The last ingredient required to use both the RK and EK method for eigenvalue problems is a way to extract eigenvalue estimates from the subspace $\mathcal{K}_{m+1}^{\text{rat}}(A, \boldsymbol{v}, \Xi)$ via the RK recurrence (6).

Similar to Arnoldi's method, we will use the Ritz pairs for this purpose. Let $(\vartheta_k, \boldsymbol{z}_k)$ satisfy,

$$\underline{K}_m^\dagger \underline{L}_m \, \boldsymbol{z}_k = \vartheta_k \boldsymbol{z}_k, \tag{8}$$

then $(\vartheta_k, \boldsymbol{x}_k = V_{m+1}\underline{K}_m \boldsymbol{z}_k)$ is called a Ritz pair. The Ritz pairs satisfy the Galerkin condition: $A\boldsymbol{x}_k - \vartheta_k \boldsymbol{x}_k \perp \mathcal{K}_m(A, q_m(A)^{-1}\boldsymbol{v})$ [29]: the residual is orthogonal to an m dimensional subspace of the rational Krylov subspace of dimension $m+1$. We call the matrix $\underline{K}_m^\dagger \underline{L}_m$ the *Galerkin projection* of the original problem A onto the rational Krylov subspace $\mathcal{K}_m(A, q_m(A)^{-1}\boldsymbol{v})$. If the last pole ξ_m is at infinity, the Ritz pairs can be computed equivalently by solving,

$$L_m \, \boldsymbol{z}_k = \vartheta_k K_m \, \boldsymbol{z}_k,$$

which avoids the computation of $\underline{K}_m^\dagger \underline{L}_m$. This small-scale generalized eigenvalue problem can be solved, for example, with the rational QZ method [26].

The convergence of the Ritz pairs can be monitored inexpensively without invoking operations with A. The residual vector $\boldsymbol{r}_k$ of the Ritz pair $(\vartheta_k, \boldsymbol{x}_k)$ is given by,

$$\begin{aligned}
\boldsymbol{r}_k &= A\boldsymbol{x}_k - \vartheta_k \boldsymbol{x}_k \\
&= AV_{m+1}\underline{K}_m \boldsymbol{z}_k - \vartheta_k V_{m+1}\underline{K}_m \boldsymbol{z}_k \\
&= V_{m+1}\underline{L}_m \boldsymbol{z}_k - V_{m+1}\underline{K}_m \underline{K}_m^\dagger \underline{L}_m \boldsymbol{z}_k \\
&= V_{m+1}(I - P)\underline{L}_m \boldsymbol{z}_k,
\end{aligned}$$

where $P = \underline{K}_m \underline{K}_m^\dagger$ is the orthogonal projector on $\mathcal{R}(\underline{K}_m)$ and $I - P$ is the orthogonal projector on the nullspace of $\underline{K}_m^*$. This expression simplifies to $\boldsymbol{r}_k = \ell_{m+1,m} \boldsymbol{v}_{m+1} \boldsymbol{e}_m^T \boldsymbol{z}_k$ if the last pole ξ_m is chosen at infinity. This means that $\|\boldsymbol{r}_k\|$ will be small if either $|\ell_{m+1,m}|$ is small, meaning that the RK subspace is nearly an invariant subspace, or if the last component of the eigenvector $\boldsymbol{z}_k$ becomes small.

The eigenvalues of interest can often be extracted from $\mathcal{K}_{m+1}^{\text{rat}}(A, \boldsymbol{v}, \Xi)$ for $m \ll N$ and m can often be much smaller than in the polynomial Krylov case if a good choice of poles Ξ is made. However, in some cases, due to storage constraints or for limiting orthogonalization costs, it is preferable to restrain m to a certain maximal dimension. If not all eigenvalues of interest have converged at this maximal m, even with a good choice of Ξ, it is necessary to restart the iteration. We develop a method to carry out a restart of the RK iteration in the next three sections. The method implicitly applies a polynomial filter to the subspace by acting on a QR-factorized representation of the RK Hessenberg pair and we show that the *condensed* QZ method of Vandebril & Watkins is a special case of this method.

3. Core transforming the rational Krylov method.

A core transformation C_i, acting on two consecutive rows i and $i{+}1$, is the embedding of a nonsingular $2{\times}2$ matrix at rows and columns i and $i{+}1$ of the identity matrix. All core transformations considered in this paper are unitary and can, for example, be a Givens rotation matrix or a Householder reflector. Core transformations exhibit many properties that make them suitable for stable and efficient computations. Since we restrict ourselves to the unitary case, the inner product of two vectors is preserved upon multiplication with C_i. The inverse of a core transformation, C_i^*, is again a core transformation. Left multiplication with a core transformation C_i only affects rows i and $i{+}1$ of a matrix, while right multiplication with the same core transformation only affects columns i and $i{+}1$. A commonly used graphical representation for core transformations is by means of a double-sided arrow pointing to the rows or columns on which it acts. Consider the following basic example:

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

In this example, the core transformation introduces a zero in position $(2, 1)$ of the matrix, bringing it to upper triangular form.

3.1. Factorizing the RK recurrence.

The QR factorization of an upper Hessenberg matrix $\underline{H}$ admits a representation with a *descending* pattern of core transformations. An example is shown in Figure 1.

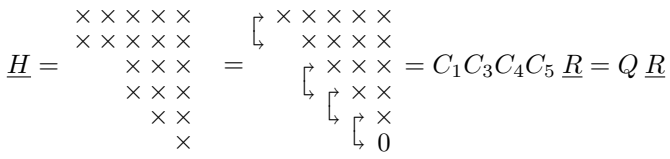


Figure 1: QR factorization of a reducible Hessenberg matrix $\underline{H}$ represented as a descending sequence of core transformations.

If the upper Hessenberg matrix $\underline{H}$ is unreduced, the pattern of core transformations becomes *condensed*. In this context, condensed means that there is exactly one core transformation acting on every pair of consecutive rows i and i+1. The example shown in Figure 1 is reducible as there is no core transformation C_2 acting on rows 2 and 3.

We can generalize the notion of a condensed QR factorization to a matrix pair. As was mentioned earlier, the EK Hessenberg pair $(\underline{L}_m, \underline{K}_m)$ is in condensed form. In terms of core transformations, this means that the unitary matrices from the QR factorizations $\underline{L}_m = Q_L \, \underline{R}_L$ and $\underline{K}_m = Q_K \, \underline{R}_K$ have the property that $Q_L \, Q_K = C_{i_1} \cdots C_{i_m}$, with $i_1, \ldots, i_m$ a permutation of $1, \ldots, m$, i.e. in the QR factorization of $(\underline{L}_m, \underline{K}_m)$ there is exactly one core transformation C_i at every row i of the EK Hessenberg pair.

An example of a factorized EK Hessenberg pair with m=5 is presented in the top row of Figure 2. The total number of core transformations for the matrix pair is equal to m=5. If the pole ξ_i is 0, the core transformation C_i is at the side of $\underline{K}_m$, for a pole at ∞, the core transformation appears at the side of $\underline{L}_m$.

The factorized representation of a general RK Hessenberg pair $(\underline{L}_m, \underline{K}_m)$ with m=5 is shown in the bottom row of Figure 2. In this case, both Hessenberg matrices admit a QR factorization with m core transformations in a descending sequence. This brings the total number of core transformations for the RK Hessenberg pair up to $2m$.

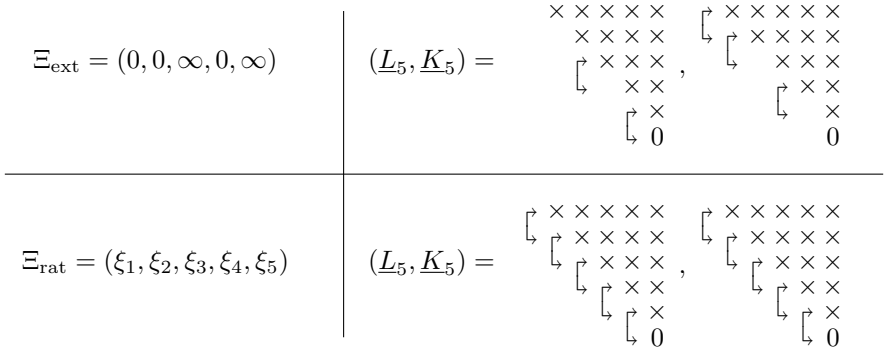


Figure 2: QR factorized representation of an EK Hessenberg pair with $m = 5$ (*top row*) and an RK Hessenberg pair (*bottom row*).

We denote the RK recurrence (6) with the RK Hessenberg pair in factorized format as,

$$A\,V_{m+1}\,\underbrace{U_1 \ldots U_m}_{Q_K}\,\underline{R}_K = V_{m+1}\,\underbrace{W_1 \ldots W_m}_{Q_L}\,\underline{R}_L, \tag{9}$$

with U_i the core transformations related with $\underline{K}_m$ and W_i those linked to $\underline{L}_m$. In this representation the ith pole of the decomposition is equal to:

$$\xi_i = \frac{(W_i)_{2,1}(R_L)_{i,i}}{(U_i)_{2,1}(R_K)_{i,i}}, \tag{10}$$

since the right-hand side is the ratio of the ith elements on the first subdiagonal of $\underline{L}$ and $\underline{K}$. In the extended Krylov setting either U_i or W_i is a diagonal matrix such that the element $2,1$ is zero and Equation (10) results in zero or infinity.

The QR factorization of the RK Hessenberg pair can be computed after the RK iteration of Algorithm 1 has been computed up to full dimension m. It can also be computed and updated iteratively during the course of Algorithm 1. This can be done as follows: assume that at step $j-1$ of Algorithm 1 we have a factorized representation,

$$\underline{K}_{j-1} = U_1 \ldots U_{j-1}\,\underline{R}_K, \quad \text{and} \quad \underline{L}_{j-1} = W_1 \ldots W_{j-1}\,\underline{R}_L,$$

available. In step j of Algorithm 1, the factorized recurrence matrices need to be updated with the new columns,

$$\underline{\boldsymbol{k}}_j = \alpha_j \underline{\boldsymbol{h}}_j - \gamma_j \begin{bmatrix} \boldsymbol{t}_j^T & 0 \end{bmatrix}^T, \quad \text{and} \quad \underline{\boldsymbol{\ell}}_j = -\beta_j \underline{\boldsymbol{h}}_j + \delta_j \begin{bmatrix} \boldsymbol{t}_j^T & 0 \end{bmatrix}^T,$$

in line 8. In order to do so, we first compute,

$$\tilde{\underline{\boldsymbol{k}}}_j = U_{j-1}^* \ldots U_1^*\,\underline{\boldsymbol{k}}_j, \quad \text{and} \quad \tilde{\underline{\boldsymbol{\ell}}}_j = W_{j-1}^* \ldots W_1^*\,\underline{\boldsymbol{\ell}}_j. \tag{11}$$

The new core transformations U_j and W_j can now be computed from the last two entries of respectively $\tilde{\underline{\boldsymbol{k}}}_j$ and $\tilde{\underline{\boldsymbol{\ell}}}_j$. Afterwards, the last entries are zeroed and the upper triangular matrices $\underline{R}_K$ and $\underline{R}_L$ can be updated with new columns $\tilde{\underline{\boldsymbol{k}}}_j$ and $\tilde{\underline{\boldsymbol{\ell}}}_j$.

The update of the factorization in step j requires $\mathcal{O}(j)$ operations, the main computational factor being the application of previous core transformations to compute $\tilde{\underline{\boldsymbol{k}}}_j$ and $\tilde{\underline{\boldsymbol{\ell}}}_j$. The total additional computational complexity to factorize $(\underline{L}_m, \underline{K}_m)$ is $\mathcal{O}(m^2)$ which is negligible in comparison with the $\mathcal{O}(m^2 N)$ orthogonalization cost in Algorithm 1.

3.2. Three operations on core transformations.

We will make use of three types of elementary operations on core transformations to formulate the methods for the implicit restart of the RK method. These operations are called the *transfer*, *fusion* and *turnover* operations. The transfer of a core transformation from left to right, or vice versa, through an upper triangular matrix is shown in Figure 3.

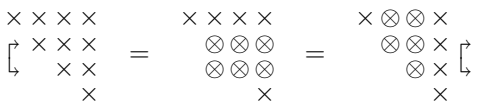


Figure 3: Transfer of a core transformation from the left of an upper triangular matrix to the right or vice versa.

Elements of the upper triangular matrix that are altered during the transfer from left to right are indicated with $\otimes$ in Figure 3. The core transformation on the left is different from the one on the right but its index is not changed and the upper triangular shape is preserved. The computational complexity of a transfer operation is $\mathcal{O}(n)$, with n the dimension of the upper triangular matrix.

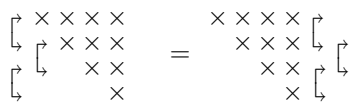
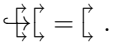


Figure 4: The ordering of core transformations is preserved under the transfer operation.

Notice that a core transformation C_i on the right of an upper triangular acts on *columns* instead of rows, as is indicated in the figure. If multiple core transformations are present in a given pattern or shape, for example the descending pattern of a Hessenberg matrix, then the complete pattern of transformations can be transferred through the upper triangular matrix. This operation preserves the mutual ordering of the core transformations, an example is shown in Figure 4.

Two core transformations that act consecutively on the same rows or columns can be multiplied and the result is again a core transformation. This is called a fusion of core transformations and is depicted as:



The turnover of a *V-shaped* pattern of 3 core transformations is given by is shown in Figure 5. This operation flips a factorization of 3 core transformations that act on rows $(i, i{+}1)$, $(i{+}1, i{+}2)$, $(i, i{+}1)$ into a factorization acting on rows $(i{+}1, i{+}2)$, $(i, i{+}1)$, $(i{+}1, i{+}2)$ or vice versa. A turnover is always possible in the unitary case. This can be proven by considering two variants for factorizing a unitary $3{\times}3$ matrix [30]. The computational complexity of both the fusion and turnover operation is $\mathcal{O}(1)$.

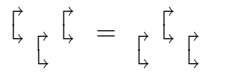


Figure 5: Turnover of a V-shaped pattern of core transformations.

Finally, we remark that two core transformation C_i and C_j commute if $|i - j| > 1$. As a consequence the mutual ordering in a pattern of core transformations is not necessarily unique.

3.3. Structure in the Galerkin projection.

We can use the transfer operation to study the structure of the Galerkin projection $\underline{K}_m^\dagger \underline{L}_m$ in Equation (8) of the original large-scale problem onto the rational Krylov subspace.

It is well-known that the Galerkin projection on a rational Krylov subspace is of a particular rank-structured from in the Hermitian case [31] and the unsymmetric case [32, 33]. We will prove this same result but via a direct proof that exploits the matrix structure instead of using the theory of orthogonal rational functions [32, 33]. Up to our knowledge this is a novel approach for this result.

Let us start with introducing the appropriate matrix structures. These are also illustrated in Figure 6. We are already acquainted with the Hessenberg matrix shown in pane I: its QR factorized representation admits a descending ordering of core transformations. This matrix structure is linked with the polynomial Krylov method as it is the structure of the Galerkin projection on the polynomial Krylov subspace. Pane II shows an example of an *extended* Hessenberg matrix. This is a relaxation of the Hessenberg matrix in the sense that the core transformations can admit any ordering. This matrix structure is linked with the extended Krylov method as it is the structure of the Galerkin projection on the extended Krylov subspace. We will show this in Lemma 3.1.

Finally, pane III shows an example of an extended Hessenberg plus diagonal matrix. Lemma 3.1 shows that this matrix structure is linked with the projection on a rational Krylov subspace. For this reason, we use the alternative name *rational Hessenberg* matrix for this matrix structure.

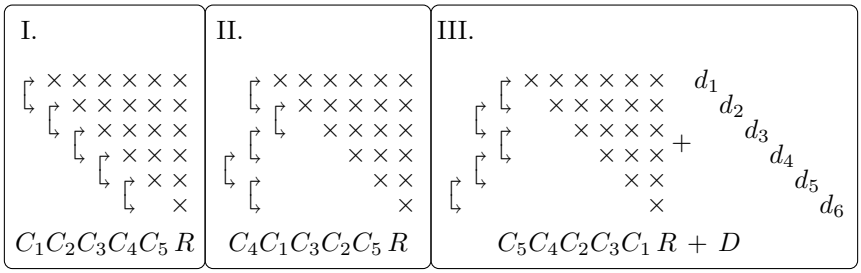


Figure 6: Examples of the representation of (I) Hessenberg, (II) extended Hessenberg and (III) rational Hessenberg matrices.

Lemma 3.1. Given a rational Krylov triplet $(V_{m+1}, \underline{K}_m, \underline{L}_m)$ corresponding to the rational Krylov subspace $\mathcal{K}^{\text{rat}}_{m+1}(A, \boldsymbol{v}_1, \Xi = (\xi_1, \ldots, \xi_m))$. Consider the vector $\boldsymbol{d} \in \mathbb{C}^m$ with $d_i = \xi_i$ for $\xi_i \neq \infty$, otherwise d_i can be any scalar. Then we have that $\underline{K}_m^\dagger \underline{L}_m$ is a rational Hessenberg matrix $QR+D$ with:

- $Q = C_{k_1} \ldots C_{k_{m-1}}$ satisfying $\begin{bmatrix} C_i\, C_{i+1} & \text{if} \quad \xi_{i+1} = \infty \\ C_{i+1}\, C_i & \text{if} \quad \xi_{i+1} \neq \infty \end{bmatrix}$, $i = 1, \ldots m-2$,
- $D = \text{diag}(\boldsymbol{d})$.

Proof. Consider the matrix $\underline{G}_m = \underline{L}_m - \underline{K}_m D$, which is an $(m+1)\times m$ upper Hessenberg matrix as both $\underline{L}_m$ and $\underline{K}_m D$ are upper Hessenberg of size $(m+1)\times m$. The subdiagonal elements of $\underline{G}_m$ are $h_{i+1,i} = \ell_{i+1,i} - d_{ii} k_{i+1,i}$ for $i \in \{1, \ldots, m\}$. As $d_{ii} = \xi_i = \ell_{i+1,i}/k_{i+1,i}$ if $k_{i+1,i} \neq 0$, we have that $h_{i+1,i} = 0$ if the pole ξ_i is not at infinity. For poles at infinity, $h_{i+1,i} = \ell_{i+1,i} \neq 0$ since there is no breakdown. The matrix $\underline{G}_m$ has thus a zero subdiagonal element whenever $\underline{K}_m$ has a nonzero subdiagonal element and vice versa. We have $\underline{K}_m^\dagger \underline{L}_m = \underline{K}_m^\dagger \underline{G}_m + \underline{K}_m^\dagger \underline{K}_m D$. As $\underline{K}_m$ is full rank, $\underline{K}_m^\dagger \underline{K}_m = I_m$ and it remains thus to examine the QR factorization of $\underline{K}_m^\dagger \underline{G}_m$ to prove the Lemma.

We get,

$$\underline{K}_m^\dagger \underline{G}_m = \underline{R}_K^\dagger Q_K^* Q_G \underline{R}_G = \begin{bmatrix} R_K^{-1} & 0 \end{bmatrix} Q_K^* Q_G \begin{bmatrix} R_G \\ 0 \end{bmatrix},$$

where R_K^{-1} is well-defined since $\underline{K}_m$ is of maximal rank. The unitary matrices can be represented as a product of core transformations as,

$$Q_K = \tilde{C}_{i_1}^* \cdots \tilde{C}_{i_k}^*, \text{ and } Q_G = \tilde{C}_{j_1} \cdots \tilde{C}_{j_\ell},$$

where $k+\ell = m$, $i_1 < i_2 < \ldots < i_k$, $j_1 < j_2 < \ldots < j_\ell$ and $\{i_1, \ldots, i_k\} \cup \{j_1, \ldots, j_l\}$ equal to $\{1, \ldots, m\}$. The Hermitian conjugates in Q_K are only introduced for convenience. The product of both unitary matrices, $\tilde{Q} = Q_K^* Q_G$ is equal to:

$$\tilde{Q} = Q_K^* Q_G = \tilde{C}_{i_k} \ldots \tilde{C}_{i_1} \tilde{C}_{j_1} \ldots \tilde{C}_{j_l} = \tilde{C}_{k_1} \cdots \tilde{C}_{k_{n-1}}.$$

We will prove next that the mutual ordering of $\tilde{C}_i$ and $\tilde{C}_{i+1}$ in the factorization of $\tilde{Q}$ is imposed by ξ_{i+1} as specified in the formulation of the Lemma.

- If $\xi_{i+1} = \infty$, then $i+1 \in \{j_1, \ldots, j_l\}$ as it was designed to create a zero in $\underline{G}_m$. There are two possibilities, either $\tilde{C}_i$ appears in Q_K^* or in Q_G. If it is in Q_K^* it is clearly to the left, if it is in Q_G, then $j_1 < \ldots < j_l$ ensures that is located to the left of C_{i+1}.

- If $\xi_{i+1} \neq \infty$, then $i{+}1 \in \{i_1, \ldots, i_k\}$ and an analogous reasoning shows that $\tilde{C}_i$ must be positioned right of $\tilde{C}_{i+1}$.

There are two possibilities for the mth core transformation. If $\xi_m \neq \infty$ then the mth core transformation in $\tilde{Q}$ is located on the left of core transformation $m-1$ and we can write $\tilde{Q} = C_m \tilde{Q}_{1\ldots m-1}$ with $\tilde{Q}_{1\ldots m-1}$ the unitary matrix formed by the first $m-1$ core transformations. This gives:

$$\underline{K}_m^\dagger \underline{G}_m = \begin{bmatrix} R_K^{-1} & 0 \end{bmatrix} C_m \tilde{Q}_{1\ldots m-1} \begin{bmatrix} R_G \\ 0 \end{bmatrix} = \begin{bmatrix} \tilde{R} & \otimes \end{bmatrix} \tilde{Q}_{1\ldots m-1} \begin{bmatrix} R_G \\ 0 \end{bmatrix} = Q_{1\ldots m-1} R.$$

In the second equality C_m is applied to columns m and $m+1$ of $\underline{R}_K^\dagger$, this preserves the upper triangular structure in the left $m \times m$ block. For the third equality $\tilde{Q}_{1\ldots m-1}$ is transferred from the right of $[\tilde{R} \otimes]$ to the left. Since $\tilde{Q}_{1\ldots m-1}$ only affects the first m columns, both the upper triangularity in the left $m \times m$ block and the mutual ordering of the core transformations are preserved.

Similarly, if $\xi_m = \infty$, the mth core transformation is located right from core transformation $m-1$ and we can write $\tilde{Q} = \tilde{Q}_{1\ldots m-1} C_m$ to get:

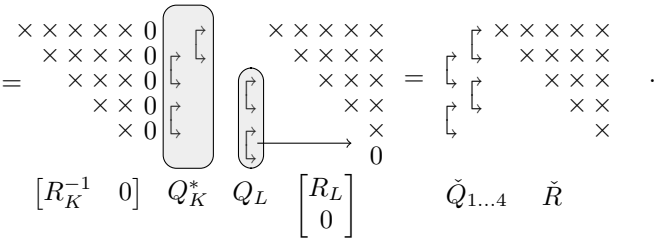
$$\underline{K}_m^\dagger \underline{G}_m = \begin{bmatrix} R_K^{-1} & 0 \end{bmatrix} \tilde{Q}_{1\ldots m-1} C_m \begin{bmatrix} R_G \\ 0 \end{bmatrix} = \begin{bmatrix} R_K^{-1} & 0 \end{bmatrix} \tilde{Q}_{1\ldots m-1} \begin{bmatrix} \tilde{R} \\ \otimes \end{bmatrix} = Q_{1\ldots m-1} R.$$

□

Let us further illustrate this structure with some simple examples.

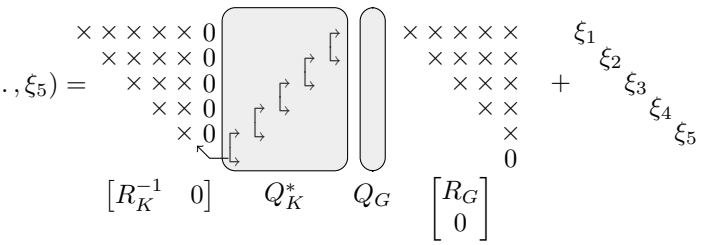
Example 3.2. In the case of the projection on an extended Krylov subspace, the vector $\boldsymbol{d}$ can be chosen as the zero vector according to Lemma 3.1 and we end up with a matrix in extended Hessenberg format. For example, the Galerkin projection $\underline{K}_m^\dagger \underline{L}_m$ on the extended Krylov subspace of Figure 2 with $\Xi_{\text{ext}} = (0, 0, \infty, 0, \infty)$ is of the form:

$\underline{K}_5^\dagger \underline{L}_5 =$

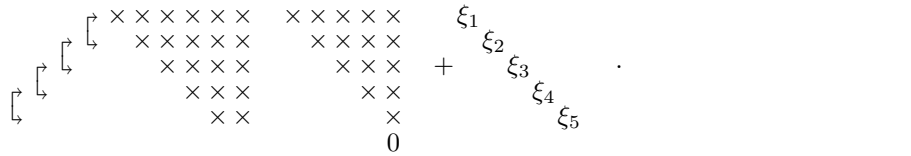


The Galerkin projection $\underline{K}_m^\dagger \underline{L}_m$ on the rational Krylov subspace of Figure 2 with $\Xi_{\text{rat}} = (\xi_1, \xi_2, \xi_3, \xi_4, \xi_5)$, assuming all $\xi_i \neq \infty$, is of the form:

$\underline{K}_5^\dagger \underline{L}_5 = \underline{K}_5^\dagger \underline{G}_5 + \operatorname{diag}(\xi_1, \ldots, \xi_5) =$



$=$



Since there are no infinite poles in this example the matrix $\underline{G}_5$, as used in the proof of Lemma 3.1, is an upper triangular matrix. As a consequence, all 5 core transformations are located in an ascending pattern that forms Q_K^*. In the second equality, the core transformation acting on columns 5 and 6 is merged into the upper triangular matrix. This creates nonzero elements in the last column, but the remaining core transformation can be transferred to the left because the first columns are still in upper triangular form. The result is shown on the second row. It is clear that the product of the 5×6 and 6×5 upper triangular matrices result in a 5×5 matrix in upper triangular form.

4. Manipulating poles via core transformations.

In this section we revisit two operations that can be used to change poles of an RK Hessenberg pencil. These operations are also studied by Berljafa & Güttel [22]. We reformulate these operations in terms of the QR-factorized representation of the RK Hessenberg pencil.

4.1. Changing the first pole.

Assume we have an RK recurrence in factorized format as in Equation (9) with poles Ξ. If we want to change ξ_1 to another pole $\hat{\xi}_1 = \hat{\alpha}_1/\hat{\beta}_1 \in \bar{\mathbb{C}}$, we can proceed as follows. First, compute the vector,

$$\boldsymbol{x} = \gamma \begin{bmatrix} \hat{\beta}_1 (W_1)_{1,1}(R_L)_{1,1} - \hat{\alpha}_1 (U_1)_{1,1}(R_K)_{1,1} \\ \hat{\beta}_1 (W_1)_{2,1}(R_L)_{1,1} - \hat{\alpha}_1 (U_1)_{2,1}(R_K)_{1,1} \end{bmatrix}, \tag{12}$$

where γ is any convenient scaling factor. Notice that $\boldsymbol{x}$ contains the nonzero elements of the first column of $(\hat{\beta}_1 \underline{L}_m - \hat{\alpha}_1 \underline{K}_m)$. Next, we compute a core transformation C_1 that introduces a zero in $\boldsymbol{x}$,

$$C_1 \boldsymbol{x} = \alpha \boldsymbol{e}_1. \tag{13}$$

If this core transformation C_1 is fused with U_1 to get $\hat{U}_1$ and with W_1 to get $\hat{W}_1$, then the first pole in $(\hat{\underline{L}}_m, \hat{\underline{K}}_m)$ is changed to $\hat{\xi}_1$, since:

$$(\hat{\underline{L}}_m - \hat{\xi}_1 \hat{\underline{K}}_m)\boldsymbol{e}_1 = C_1(\hat{\beta}_1 \underline{L}_m - \hat{\alpha}_1 \underline{K}_m)\boldsymbol{e}_1 = \frac{1}{\gamma} C_1 \boldsymbol{x} = \frac{\alpha}{\gamma}\boldsymbol{e}_1.$$

In other words, the first column of $(\hat{\underline{L}}_m - \hat{\xi}_1 \hat{\underline{K}}_m)$ is a scalar multiple of $\boldsymbol{e}_1$ which means that $\hat{\xi}_1$ is its first pole. This procedure in shown in Figure 7. To maintain the RK recurrence (Equation (9)), the orthonormal basis is updated as $\hat{V}_{m+1} = V_{m+1} C_1^*$. This operation only involves the first two columns of V_{m+1}.

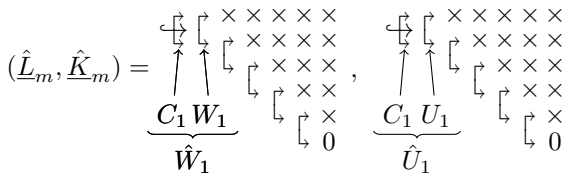


Figure 7: Procedure to change the first pole by a fusion with C_1 computed with Equations (12) and (13).

4.2. Swapping poles.

Two adjacent poles ξ_i and ξ_{i+1} in the RK Hessenberg pencil can be *swapped* by using standard methods for reordering eigenvalues in the generalized Schur decomposition [22, 26]. Details and solutions for the reordering problem can be found in Kågström & Poromaa [34] and Van Dooren [35].

In this section we show how the reordering technique can be combined with the representation in terms of core transformations. The method thus starts with a factorized RK Hessenberg pencil $(\underline{L}_m, \underline{K}_m)$ having poles $\Xi = (\xi_1, \ldots, \xi_{i-1}, \xi_i, \xi_{i+1}, \xi_{i+2}, \ldots, \xi_m)$. The end result is an equivalent factorized RK Hessenberg pencil $(\hat{\underline{L}}_m, \hat{\underline{K}}_m)$ with poles $\hat{\Xi} = (\xi_1, \ldots, \xi_{i-1}, \xi_{i+1}, \xi_i, \xi_{i+2}, \ldots, \xi_m)$. The swapping of poles ξ_i and ξ_{i+1} is achieved by acting on rows $i{+}1{:}i{+}2$ and columns $i{:}i{+}1$ of $(\underline{L}_m, \underline{K}_m)$. The relevant part of the RK Hessenberg pencil is shown in Figure 8 both in its factorized and full representation.

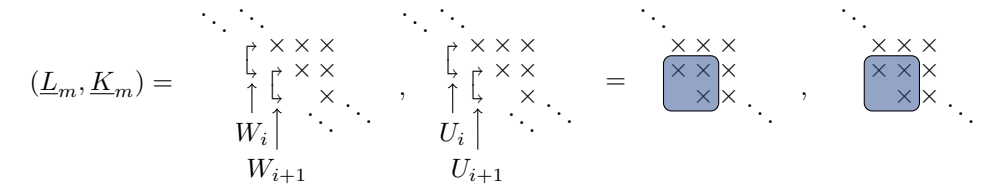


Figure 8: Rows and columns i up to $i{+}2$ of the RK Hessenberg pencil in factorized representation (*left*) and full form (*right*). The $2{\times}2$ shaded part on the right contains the adjacent poles ξ_i and ξ_{i+1}.

The shaded area in Figure 8 is the $2{\times}2$ part that contains ξ_i and ξ_{i+1}. This part is in generalized Schur form. The reordering techniques [34, 35] can thus be used to change the order of ξ_i and ξ_{i+1}. These methods compute two core transformations C_{i+1} acting on rows $i{+}1{:}i{+}2$ and C_i acting on columns $i{:}i{+}1$ such that the poles in the shaded area are swapped. These core transformations can be computed based on the factorized representation as,

$$\begin{aligned} C_{i+1}\begin{bmatrix}(W_i)_{2,1}(R_L)_{i,i} & (W_i)_{2,1}(R_L)_{i,i+1} + (W_i)_{2,2}(W_{i+1})_{1,1}(R_L)_{i+1,i+1} \\ & (W_{i+1})_{2,1}(R_L)_{i+1,i+1}\end{bmatrix} C_i &= \begin{bmatrix}\beta_{1,1} & \times \\ & \beta_{2,2}\end{bmatrix}, \\ C_{i+1}\begin{bmatrix}(U_i)_{2,1}(R_K)_{i,i} & (U_i)_{2,1}(R_K)_{i,i+1} + (U_i)_{2,2}(U_{i+1})_{1,1}(R_K)_{i+1,i+1} \\ & (U_{i+1})_{2,1}(R_K)_{i+1,i+1}\end{bmatrix} C_i &= \begin{bmatrix}\gamma_{1,1} & \times \\ & \gamma_{2,2}\end{bmatrix}, \end{aligned} \tag{14}$$

with,

$$\hat{\xi}_i = \frac{\beta_{1,1}}{\gamma_{1,1}} = \frac{(W_{i+1})_{2,1}(R_L)_{i+1,i+1}}{(U_{i+1})_{2,1}(R_K)_{i+1,i+1}} = \xi_{i+1}, \tag{15}$$

and,

$$\hat{\xi}_{i+1} = \frac{\beta_{2,2}}{\gamma_{2,2}} = \frac{(W_i)_{2,1}(R_L)_{i,i}}{(U_i)_{2,1}(R_K)_{i,i}} = \xi_i. \tag{16}$$

Here the matrices on the left-hand side of Equation (14) make up the shaded part in Figure 8 computed efficiently from the factorized representation.

If we introduce C_{i+1} on the left and C_i on the right in the factorized RK Hessenberg pencil, we get the pattern shown in Figure 9. This figure displays the same part of the RK Hessenberg pencil as Figure 8.

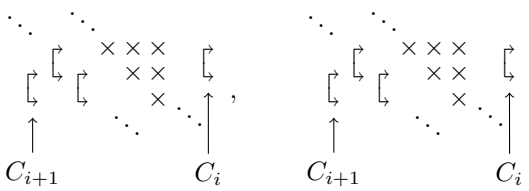


Figure 9: Introduction of C_{i+1} and C_i from Equation 14 in the factorized RK Hessenberg pair.

Merging C_i on the right into the upper triangular matrices, creates a nonzero entry in position $(i{+}1, i)$ as shown in Figure 10. Further, a turnover can be performed on the V-shaped pattern of core transformations on the left side of the matrices. This is also shown in Figure 10.

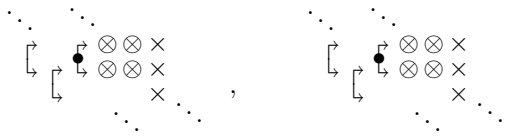


Figure 10: Merging of C_i on the right with the upper triangular matrices changes columns i and $i{+}1$ and creates a nonzero element in position $(i{+}1, i)$. The V-shaped pattern on core transformations on the right has undergone a turnover operation.

As the configuration in both Figures 9 and 10 is mathematically equivalent with Equation (14), it follows that the core transformations marked with a • will create a zero in position $(i{+}1, i)$ and restore the upper triangular form when applied to rows i and $i{+}1$ of the matrices. The end result is shown in Figure 11, which is a factorized representation of the RK Hessenberg pair where poles ξ_i and ξ_{i+1} have been swapped.

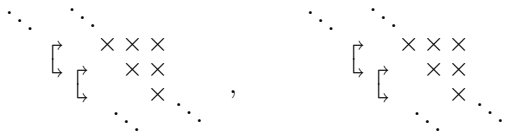


Figure 11: End result of the swap: the core transformations marked with • in Figure 10 restored the upper triangular form of the matrices.

Finally, to maintain the RK recurrence (Equation (9)), the orthonormal basis is updated as $\hat{V}_{m+1} = V_{m+1}C^*_{i+1}$.

5. An implicit QZ step for filtering the factorized rational Krylov method.

Algorithm 2 presents the implicit QZ method to apply a filter to the factorized RK recurrence of Equation (9). This is a reformulation of the method of Berljafa & Güttel [22, Section 4.3] in terms of operations on core transformations.

The gist of Algorithm 2 is simple. Starting with a factorized RK triplet $(V_{m+1}, \underline{K}_m, \underline{L}_m)$ with starting vector $\boldsymbol{v}$ and pole tuple $\Xi = (\xi_1, \ldots, \xi_m)$, Algorithm 2 first changes the pole ξ_1 to a shift ϱ in line 1. Next, this shift is swapped $m{-}1$ times until it arrives the last position. This is done in lines $2-4$. At this stage the pole tuple is $\hat{\Xi} = (\xi_2, \ldots, \xi_m, \varrho)$. The span of $\hat{V}_{m+1} = V_m\, C^*_1 \ldots C^*_m$ is not changed, $\mathcal{R}(\hat{V}_{m+1}) = \mathcal{R}(V_{m+1})$, but the vectors are rearranged. The new starting vector is given by:

$$\hat{\boldsymbol{v}} = \hat{V}_{m+1}\boldsymbol{e}_1 = V_{m+1}\, C^*_1 \ldots C^*_m \boldsymbol{e}_1 = V_{m+1}\, C^*_1 \boldsymbol{e}_1 = \hat{\gamma} V_{m+1}(\underline{L}_m - \varrho \underline{K}_m)\boldsymbol{e}_1. \tag{17}$$

The third equality used the property that C^*_i does not act on the first row and column if $i > 1$. The fourth equality used $\boldsymbol{x} = (\underline{L}_m - \varrho \underline{K}_m)\boldsymbol{e}_1 = \alpha C^*_1 \boldsymbol{e}_1$, which is based on Equations (12) and (13). The rational Krylov recurrence (Equation (6)) implies:

$$(A - \varrho I)V_{m+1}(\underline{L}_m - \xi_1 \underline{K}_m) = (A - \xi_1 I)V_{m+1}(\underline{L}_m - \varrho \underline{K}_m), \tag{18}$$

as this essentially reduces to Equation (6) after expanding and canceling the terms on the left-hand and right-hand side. Rearranging Equation (18) and combining with Equation (17) gives,

$$\begin{aligned} \hat{\boldsymbol{v}} &= \hat{\gamma} V_{m+1}(\underline{L}_m - \varrho \underline{K}_m)\boldsymbol{e}_1 = \hat{\gamma}(A - \xi_1 I)^{-1}(A - \varrho I)V_{m+1}(\underline{L}_m - \xi_1 \underline{K}_m)\boldsymbol{e}_1 \\ &= \check{\gamma}(A - \xi_1 I)^{-1}(A - \varrho I)\boldsymbol{v}. \end{aligned} \tag{19}$$

Algorithm 2 Single shift, implicit QZ step for RK `RK_QZ`

Input: Factorized RK recurrence: V_{m+1}, $\underline{K}_m = (U_1, \ldots, U_m, \underline{R}_K)$, $\underline{L}_m = (W_1, \ldots, W_m, \underline{R}_L)$, and shift: $\varrho = \mu/\nu$

Output: Filtered factorized RK recurrence: V_m, $\underline{K}_m = (U_1, \ldots, U_{m-1}, \underline{R}_K)$, $\underline{L}_m = (W_1, \ldots, W_{m-1}, \underline{R}_L)$

1: *Use the procedure of Section 4.1 to compute the initial core transformation C_1 to change ξ_1 to ϱ. Perform the fusion operation and update the RK basis matrix:*

$$U_1 \leftarrow C_1 U_1, \quad W_1 \leftarrow C_1 W_1, \quad V_{m+1} \leftarrow V_{m+1} C_1^*.$$

2: **for** $i = 1, 2, \ldots, m-1$ **do**

3: *Use the procedure of Section 4.2 to compute the left (C_{i+1}) and right (C_i) core transformations that swap pole ξ_i with ξ_{i+1}:*

a: Update the RK basis matrix:

$$V_{m+1} \leftarrow V_{m+1} C_{i+1}^*.$$

b: Merge the core transformation on the right, C_i, in the upper triangular matrices:

$$\underline{R}_K \leftarrow \underline{R}_K C_i, \quad \underline{R}_L \leftarrow \underline{R}_L C_i.$$

c: Perform the turnovers:

$$[U_i, U_{i+1}, \check{C}_i] \leftarrow \text{turnover}(C_{i+1}, U_i, U_{i+1}),$$
$$[W_i, W_{i+1}, \tilde{C}_i] \leftarrow \text{turnover}(C_{i+1}, W_i, W_{i+1}).$$

d: Merge $\check{C}_i$ into $\underline{R}_K$ to create a zero in position $(i{+}1, i)$ and merge $\tilde{C}_i$ into $\underline{R}_L$ to create a zero in position $(i{+}1, i)$.

4: **end for**

5: Reduce the order of the RK recurrence by discarding the last column of V_{m+1}, the last row and column of $\underline{R}_L$ and $\underline{R}_K$ and the last core transformations U_m and W_m.

return V_m, $\underline{K}_m = (U_1, \ldots, U_{m-1}, \underline{R}_K)$, $\underline{L}_m = (W_1, \ldots, W_{m-1}, \underline{R}_L)$

From the uniqueness of a rational Krylov recurrence [22, Theorem 3.2], it follows that the result of Algorithm 2 is a factorized RK triplet satisfying:

$$\mathcal{R}(\hat{V}_m) = \mathcal{K}_m^{\text{rat}}(A, (A - \xi_1 I)^{-1}(A - \varrho I)\boldsymbol{v}, (\xi_2, \ldots, \xi_m)).$$

An important strength of the formulation of the QZ step in terms of core transformation is that *deflation* can be monitored easily and accurately in every step of the iteration. If for any index i both core transformations U_i and W_i of the RK Hessenberg pair become equal to a diagonal matrix up to machine precision $\varepsilon_{\text{mach}}$ during any of the steps of Algorithm 2 then V_i spans an invariant subspace. Consequently the Ritz pairs of $(\underline{L}_i, \underline{K}_i)$ will be exact. More details on deflations based on core transformations can be found in Mach & Vandebril [27]. A full overview of the advantages of reformulating eigenvalue algorithms in terms of core transformations can be found in the monograph [36].

5.1. Filtering the extended Krylov method.

Algorithm 2 can be used as a filter for both the rational Krylov and the extended Krylov method as the latter is a special instance of the former. However, as the EK Hessenberg pair is in condensed form, there is additional structure that can be exploited to rephrase Algorithm 2 slightly.

The method to introduce the shift as described in Section 4.1 and line 1 of Algorithm 2 remains unchanged. After the shift has been introduced it appears as an additional core transformation in the condensed EK Hessenberg pair that can be *chased* with the same technique as used in the extended QZ method [25]. This chasing procedure can also be interpreted as a pole swap where the first pole is equal to the shift to be chased and the second pole is either at 0 or ∞ depending on which side the next core transformation is located. The situation at step i is illustrated in Figure 12: the ith pole is now equal to the shift that was introduced at the start and the $(i{+}1)$st pole is either at 0 or ∞. In Figure 12 it is not specified which matrix represents $\underline{L}_m$ and $\underline{K}_m$ as the procedure in both cases is completely similar up to a reversal of their roles.

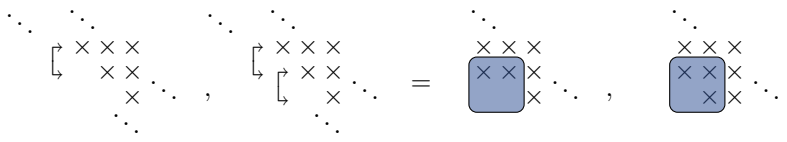


Figure 12: Pole swapping in the extended Krylov setting: the ith pole is equal to the shift ϱ, the $(i{+}1)$st pole is either 0 or ∞ which means there is only 1 core transformation with index $i{+}1$.

The core transformation that is located on the left-hand side can be transferred through the left upper triangular matrix. This is shown in the left part of Figure 13. Here it can be removed by right multiplication with its inverse. This effectively moves the Hermitian conjugate core transformation to the right-hand side of the right upper triangular matrix. There it can again be transferred to the left-hand side. This procedure is sketched in the right part of Figure 13.

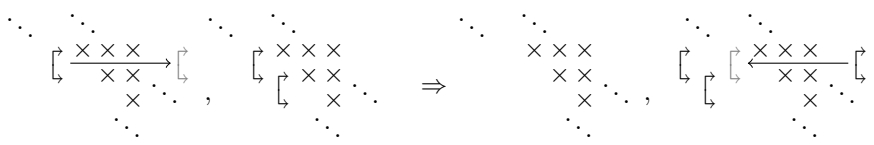


Figure 13: Transfer of the core transformation on the left-hand side of the left matrix through the upper triangular (*left*). Removal of this core transformation moves it to the matrix on the right, where it is transferred through the upper triangular (*right*).

Now the pattern on the right-hand side is ready for a turnover. This operation has been executed in the left side of Figure 14. After the turnover, the leftmost core transformation is *free* as it can be moved to the matrix on the left-hand by left multiplication with its Hermitian conjugate. This is shown in the right side of the Figure 14.

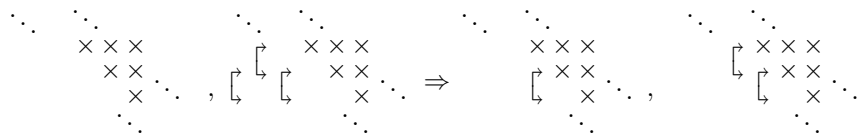


Figure 14: Turnover of the V-shaped pattern (*left*). Removal of the free core transformation on the right matrix moves it to the left matrix (*right*).

This completes the swap in the case of a condensed matrix pair. As this entire procedure is in fact an equivalence transformation applied to the shaded part indicated in the Figure 12, it preserves the eigenvalues, which are poles ξ_i and ξ_{i+1} of the EK Hessenberg pair, and as such the shift has been swapped one position down.

6. Numerical experiments.

A generic approach for a restarted rational Krylov iteration is listed in Algorithm 3. This algorithm leaves open two major questions.

The first question is an approach to select the poles during the expansion phase. The rational Krylov method allows for plenty of freedom in this respect. If one has no a priori knowledge about the problem at hand, the extended Krylov method can be a good alternative as it contains fewer parameters. This is especially true during the first iterations, afterwards a motivated choice of poles might be made based on information already available. If eigenvalues in a certain region of interest are searched after, the poles can be chosen in such a way that they form a rational filter which emphasizes the eigenvalues inside the region of interest, see [37] for a detailed description and a connection with contour integration techniques. In Example 6.3 we will use this approach to determine eigenvalues inside a contour. We will not go into further detail on the problem of pole selection.

A second issue is how to pick the shifts for the filter polynomial. Different practices have been proposed in the literature. They all attempt to create a filter polynomial $p_f \in \mathcal{P}_p$ that has the property that $|p_f(z)|$ is large on Ω_{wanted} and small on Ω_{unwanted}, where Ω_{wanted} and Ω_{unwanted} are disjoint compact sets in $\mathbb{C}$. A first method is the use of *exact* shifts [17]. These are the p Ritz values that are most distant from Ω_{wanted}. Another option is to use shifts as the zeros of Chebyshev polynomials on an ellipse [3, 17]. The use of *Leja* shifts, proposed in [38, 39], is a third possibility.

Example 6.1. In the first experiment, we use Algorithm 3 to determine the rightmost eigenvalues of a small test problem using extended and rational Krylov subspaces and exact shifts for the filter. The exact shifts are the leftmost Ritz values. We consider a matrix $A \in \mathbb{R}^{102\times 102}$ which is nonzero in the first 100 diagonal entries and in the last 2×2 block only. The diagonal entries are equal to $-100, -99, \ldots, -1$ and the 2×2 block leads to the complex conjugate pair of eigenvalues $\pm 25i$. This construction mimics the physical situation in the double-diffusive convection example [40, 41]. The spectrum of A is shown in Figure 15.

The rightmost eigenvalues of this matrix are $\pm 25i$. Assume we can only store a maximum of m=8 basis vectors in memory. For the restart phase we choose the parameter p=6. The starting vector is $[1 \cdots 1]^T$ and the iteration is repeated until the complex conjugate pair of Ritz values has converged to an error smaller than 10^{-8}.

Figure 16 shows the convergence of the desired Ritz values for 3 different options of poles Ξ. The error $|\lambda_{1,2} - \vartheta|/|\lambda_{1,2}|$ is shown in function of the dimension of the subspace. The left pane

Algorithm 3 Restarted rational Krylov algorithm RESTARTED_RK

Input: $A \in \mathbb{C}^{N\times N}$, $\mathbf{0} \neq \boldsymbol{v} \in \mathbb{C}^N$, maximal subspace dimension m, restart length p, number of desired Ritz pairs l $(p + l \leq m)$

Output: $\{(\vartheta_k, \boldsymbol{x}_k)\}_{k=1}^{l}$

1: Start:
   a: Select poles Ξ_m
   b: $[V_{m+1}, \underline{K}_m, \underline{L}_m] \leftarrow \texttt{RK}(A, \boldsymbol{v}, \Xi_m)$ ▷ Algorithm 1 in combination with Section 3.1.
   c: Check convergence of l *most* desired Ritz pairs $\{(\vartheta_k, \boldsymbol{x}_k)\}_{k=1}^{l}$
2: **while** not converged **do**
3: Select p shifts $(\varrho_k)_{k=1}^{p}$
4: **for** $j = 1 \ldots p$ **do**
5: $[V_{m-j+1}, \underline{K}_{m-j}, \underline{L}_{m-j}] \leftarrow \texttt{RK_QZ}(V_{m-j+2}, \underline{K}_{m-j+1}, \underline{L}_{m-j+1}, \varrho_j)$ ▷ Algorithm 2.
6: **end for**
7: Select $m - p$ new poles Ξ_{m-p}
8: Expand: $[V_{m+1}, \underline{K}_m, \underline{L}_m] \leftarrow \texttt{RK}(A, V, \underline{K}, \underline{L}, \Xi_{m-p})$ ▷ Algorithm 1 in combination with Section 3.1.
9: Check convergence of l *most* desired Ritz pairs $\{(\vartheta_k, \boldsymbol{x}_k)\}_{k=1}^{l}$
10: **end while**



Figure 15: Spectrum of the problem of size 102 in Example 6.1

shows the result for $\Xi_1 = (0, 0, \ldots)$, meaning that only operations with A^{-1} are used and we have an extended Krylov subspace. The restart is performed with the EK swap of Section 5.1. We observe that the convergence for the complex conjugate pair is slow and 5 restarts are required to meet the convergence criterion. The middle pane shows the convergence for $\Xi_2 = (\infty, \infty, \ldots)$, a polynomial Krylov subspace. The convergence is much faster in this case and only 3 restarts are required since the error is reduced by approximately 2 two orders of magnitude after every restart.

Considering the spectrum of A, this is what one would expect. The complex conjugate pair of eigenvalues is situated at $\pm 0.04i$ for A^{-1}. They lie in the cluster of eigenvalues near zero and are not well separated. This has a large impact on the convergence of the method. In the spectrum of the original matrix A the complex conjugate pair of eigenvalues is well separated. Hence the more rapid convergence with Ξ_2.

The right pane displays the result for a fully rational pole strategy. The initial subspace is constructed using the RK pole tuple $(-70.5, -60.5, \ldots, -10.5)$ with poles along the negative real axis. As this does not lead to significant convergence, the pole tuple is changed to $(22i, -22i, 16i, -16i, 10i, -10i)$ after the first restart. These poles along the imaginary axes speed up the convergence and only two restarts are required with this strategy.

Example 6.2. We study the benchmark problem from Elman et al. (2012)[42]. This problem also stems from fluid dynamics and is a model for the flow in a unit-square cavity with the lid moving from left to right. The Q_2 — Q_1 finite element discretization with IFISS from Elman et al. (2014)[43] resulted in a generalized eigenvalue problem $(A, B) \in \mathbb{R}^{9540\times 9540}$. The Reynolds number Re is 7800 for the pencil we consider. The critical Reynolds number of this problem Re_c is slightly less than 7929 [42, 44]. The pencil we consider is thus stable.

Both matrices A and B of the matrix pencil (A, B) are nonsingular such that we can apply an EK method for the generalized eigenvalue problem. The extended Krylov method for matrix pairs is derived by replacing I with B in the Möbius transformation (4). This leads to operations

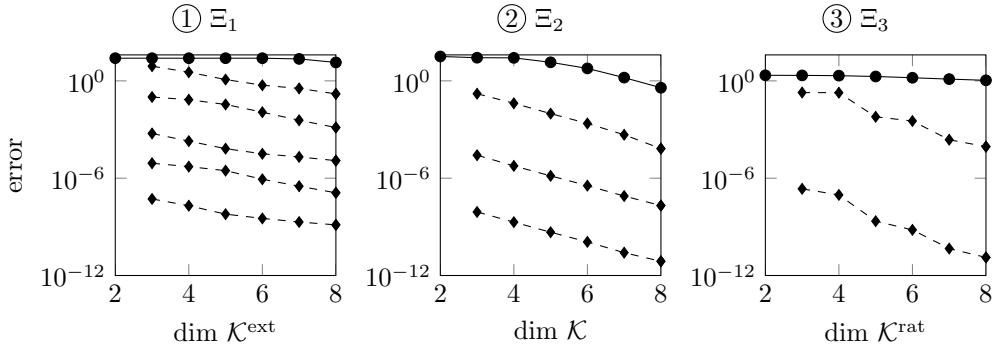
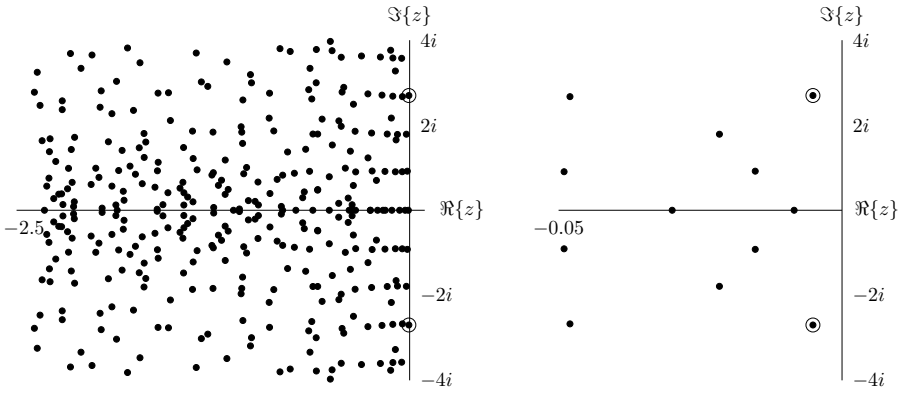


Figure 16: Convergence behavior for the restarted RK iteration for three different choices of Ξ. The convergence of the initial subspace is shown with a solid line, convergence after restarting is indicated with a dashed line. *Left:* extended, *Middle:* polynomial, *Right:* rational Krylov iteration.

with $A\,B^{-1}$ for poles at ∞ and with $B\,A^{-1}$ for poles at 0. The LU factorization of A takes 142 s in MATLAB and 27 s for B on an Intel Xeon CPU E5-2697. It is feasible to factorize both A and B once, but repeating this every few iterations is costly. Hence we prefer the extended Krylov method over the rational Krylov method.

Figure 17 shows the spectrum of (A, B). The left pane shows 343 eigenvalues in a region of the complex plane near the imaginary axis. The rightmost eigenvalues of (A, B) appear in the complex conjugate pair $\lambda_{1,2} = -0.005135 \pm 2.698447i$. They are encircled in Figure 17. The right pane provides a closeup of the region near $\lambda_{1,2}$.



(a) 343 eigenvalues of (A, B) (b) 12 eigenvalues of (A, B)

Figure 17: The spectrum of the driven cavity problem. The encircled eigenvalues are the rightmost eigenvalues.

Table 1 lists the results for 3 different experiments with 3 different choices of Ξ_{ext}. These are cyclic pole tuples and the first column of Table 1 lists the first cycle in Ξ_{ext}. The ratio of poles at 0 decreases from the first to the third row as is indicated in the second column which shows the ratio of operations with poles at 0 with the total number of operations. The third column gives the requested tolerance for convergence, the fourth the number of restarts and the last column the residual of the rightmost Ritz values.

The residual is in this case evaluated as,

$$\frac{\|Ax - \lambda Bx\|_\infty}{\|A\|_\infty + |\lambda| + \|B\|_\infty}.$$

In all three experiments, we create a subspace of dimension m=100, which is then reduced with p=50 exact (leftmost) shifts during the restart. Since this is a rather 'difficult' problem, the dimen-

sion of the subspace is kept comparatively large. In order to retrieve the rightmost eigenvalues, the convergence criterion is applied to the 12 rightmost Ritz values. If all 12 have a residual less than the tolerance, the algorithm is halted. The tolerance is adjusted for each Ξ_{ext} in such a way that a good result is achieved within a reasonable number of restarts.

The results indicate that retrieving the rightmost eigenvalues of this problem up to good accuracy is feasible with a small number of restarts. The first two choices of Ξ_{ext} give a significantly faster convergence than the third. When the tolerance in the third experiment is lowered to 10^{-8}, the method fails to converge in a reasonable number of restarts. We conclude that for this problem it is beneficial to include operations with pole at 0.

The ARPACK [45] implementation of implicitly restarted Arnoldi, which is available in MATLAB as the command `eigs`, did not retrieve the rightmost eigenvalues. This experiment demonstrates that the extended Krylov method can sometimes be a suitable choice for finding a few eigenvalues of a matrix if two conditions are satisfied. First, the convergence of the polynomial Krylov method is too slow to find the eigenvalues of interest within a reasonable number of restarts and with subspaces of small enough dimensions. Second, the computational cost of computing an LU factorization of the matrix is too large to repeat every few iterations, which excludes the rational Krylov method as a viable option, but it is small enough to do once. This second condition leaves both the extended Krylov method and shift-and-invert Arnoldi as suitable options since they both require only one matrix factorization.

Ξ_{ext}						$\frac{\#BA^{-1}\text{op.}}{\text{all op.}}$	tolerance	restarts	residual norm
∞	0	0	0	0	$\cdots$	4/5	$3 \cdot 10^{-10}$	12	$9.3 \cdot 10^{-12}$
∞	0	0	0	$\cdots$		3/4	$8 \cdot 10^{-10}$	9	$2.7 \cdot 10^{-11}$
∞	0	0	$\cdots$			2/3	$1.5 \cdot 10^{-8}$	6	$1.4 \cdot 10^{-8}$

Table 1: Summary of the results of Algorithm 3 on the cavity flow model with m=100, p=50 and $v = [1 \cdots 1]^T$ with three different options of Ξ_{ext}. The convergence is checked for the 12 rightmost Ritz values. The first column specifies the first cycle of Ξ_{ext}, the second column lists the fraction of poles at 0 in Ξ_{ext}, the third column gives the requested tolerance, the fourth column the number of restarts and the last column the residual norm upon convergence.

Example 6.3. In this last example, we make a direct comparison between the results obtained with Algorithm 3 and the explicit QZ step of De Samblanx et al. which is listed in Algorithm 4. In line 2 of Algorithm 4 an orthogonal matrix $Z \in \mathbb{C}^{m \times m-1}$ is computed for which the vector $(\nu \underline{L}_m^* + \mu \underline{K}_m^*)\boldsymbol{q}$ is in the nullspace of Z^*. This condition is not restrictive and does not define Z uniquely. Two choices for Z are used in our experiment: Z_1 as computed by Algorithm 6.1 of [21] and Z_2 computed from the full QR factorization $\begin{bmatrix} \boldsymbol{z} & Z_2 \end{bmatrix} \begin{bmatrix} \alpha \\ 0 \end{bmatrix} = (\nu \underline{L}_m^* + \mu \underline{K}_m^*)\boldsymbol{q}$.

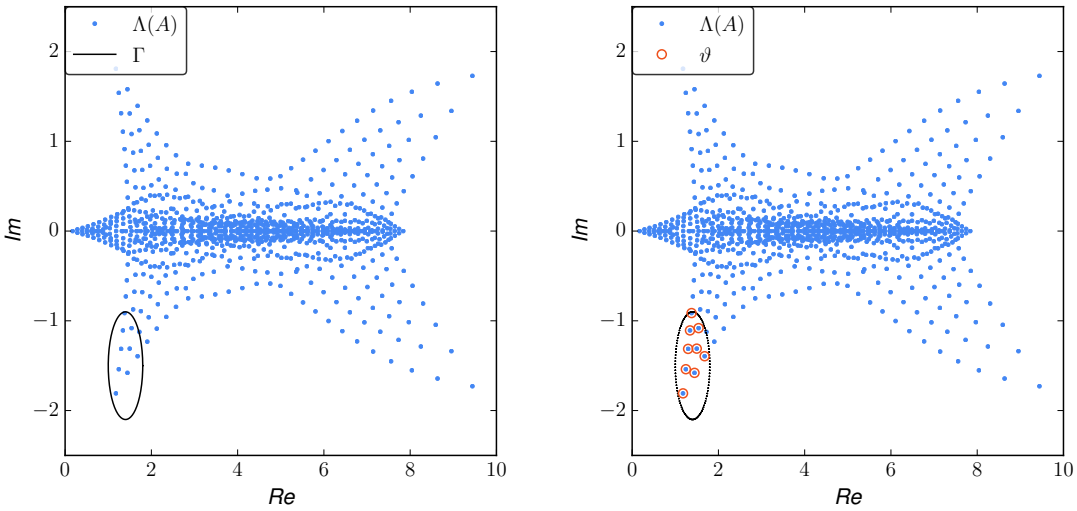
Algorithm 4 Single shift, explicit QZ step for RK [21]

Input: V_{m+1}, $\underline{K}_m$, $\underline{L}_m$, $\varrho = (\mu, \nu)$
Output: $\check{V}_m$, $\underline{\check{K}}_{m-1}$, $\underline{\check{L}}_{m-1}$

1: Compute full QR factorization $\begin{bmatrix} \check{Q} & \boldsymbol{q} \end{bmatrix} \begin{bmatrix} R \\ 0 \end{bmatrix} \coloneqq \mu \underline{L}_m + \nu \underline{K}_m$
2: Compute Z satisfying $\boldsymbol{q}^*(\bar{\nu} \underline{L}_m + \bar{\mu} \underline{K}_m)Z = 0$
3: $\underline{\check{K}}_{m-1} \coloneqq \check{Q}^* \underline{K}_m Z$
4: $\underline{\check{L}}_{m-1} \coloneqq \check{Q}^* \underline{L}_m Z$
5: $\check{V}_m = V_{m+1} \check{Q}$

The matrix we consider is `PDE900` from the MatrixMarket collection. This is a real matrix of size 900×900. We are interested in determining the 9 eigenvalues of this matrix inside the elliptical contour Γ, shown in Figure 18(a). For this purpose, the contour is discretized with $N = 110$ points and both the poles Ξ and filter shifts ϱ are located at these discretization nodes.

For more details on this choice of rational filter and connections with contour integration methods, see Van Beeumen, Meerbergen & Michiels (2017) [37].



(a) Spectrum and continuous contour.

(b) Spectrum, discretized contour and Ritz values with implicit QZ

Figure 18: Problem setting and results with the implicit QZ method.

Given that all poles are on a contour in $\mathbb{C}$, we are dealing with a proper rational Krylov iteration. The tolerance is set to 10^{-7}. After 3 outer iterations of adding N poles and applying N filter shifts with the implicit QZ step, the problem can be deflated. The iteration found an invariant subspace containing the Ritz values of interest which are shown in Figure 18(b).

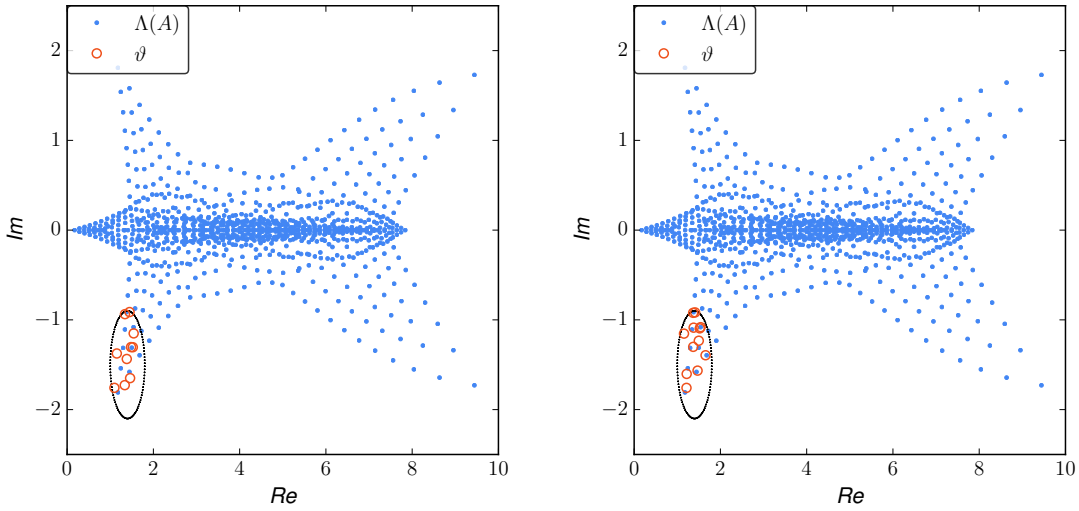
With the explicit QZ step of Algorithm 4, deflation does not occur or goes unnoticed and after the maximum of 6 outer iterations the Ritz values obtained with the choice of Z_1 are shown in Figure 19(a) and in Figure 19(b) for Z_2. Clearly, the method did not converge and the explicit QZ step distorts the information in the rational Krylov subspace.

This example demonstrates that the implicit QZ step is superior to the explicit step. Not only is it computationally more efficient, it behaves more stable and allows for accurate deflation monitoring.

7. Conclusion.

This paper presented an implicit QZ step, formulated in terms of operations on core transformations, that is useful for filtering and restarting the rational Krylov iteration. It is mathematically equivalent with the method of Berljafa & Güttel [22] but has some advantages. First, the representation with core transformations allows for accurate deflation monitoring throughout the restart phase. Second, the representation with core transformations allowed us to formulate an alternative proof that the Galerkin projection on a rational Krylov subspace has the structure of a rational Hessenberg matrix in Lemma 3.1. Third, it is in essence a reformulation of the rational QZ method [26] in terms of core transformations. In Section 5.1 we showed that this is a generalization of the extended QZ method [25] and reinterpreted the *core chasing* step as a pole swap.

In the numerical experiments, we tested the method on three different problems and demonstrated the validity of the implicit approach both for the extended and rational Krylov methods. We showed how extended Krylov can, in particular cases, be an interesting method for the computation of the rightmost eigenvalues. We compared our method with ARPACK and with the implicit restart method proposed by De Samblanx et al. [21], and showed that the new method can outperform these in some scenarios.



(a) Spectrum, discretized contour and Ritz values with explicit QZ with option Z_1.

(b) Spectrum, discretized contour and Ritz values with explicit QZ with option Z_2.

Figure 19: Results with the explicit QZ method.

Acknowledgements.

The authors thank the referee of the initial draft for the suggestion to include the rational Krylov method in the scope of the paper, and the referee of the revised manuscript for encouraging us to study the *RK-to-RK* conversion. We also thank Roel Van Beeumen for his input and for the example where the implicit method makes a significant improvement over the explicit method.

The research was partially supported by the Research Council KU Leuven, projects C14/16/056 (Inverse-free Rational Krylov Methods: Theory and Applications), OT/14/074 (Numerical algorithms for large scale matrices with uncertain coefficients).

References

[1] W. E. Arnoldi, The principle of minimized iteration in the solution of the matrix eigenvalue problem, Quart. Appl. Math. 9 (1951) 17–29.

[2] Y. Saad, Variations on Arnoldi's method for computing eigenelements of large unsymmetric matrices, Linear Algebra Appl. 34 (1980) 269–295.

[3] Y. Saad, Chebyshev acceleration techniques for solving nonsymmetric eigenvalue problems, Math. Comput. 42 (166) (1984) 567–588.

[4] Y. Saad, Numerical solution of large nonsymmetric eigenvalue problems, Comput. Phys. Commun. 53 (1-3) (1989) 71–90.

[5] Y. Saad, Numerical Methods for Large Eigenvalue Problems, Manchester University Press.

[6] A. Kuijlaars, Which eigenvalues are found by the Lanczos method?, SIAM J. Matrix Anal. Appl. 22 (1) (2000) 306–321.

[7] A. Kuijlaars, Convergence analysis of Krylov subspace iterations with methods from potential theory, SIAM Rev. 48 (1) (2006) 3–40.

[8] T. Ericsson, A. Ruhe, The spectral transformation Lanczos method for the numerical solution of large sparse generalized symmetric eigenvalue problems, Math. Comput. 35 (152) (1980) 1251–1268.

[9] B. N. Parlett, Y. Saad, Complex shift and invert strategies for real matrices, Linear Algebra Appl. 88-89 (1987) 575–595.

[10] K. Meerbergen, A. Spence, Implicitly restarted Arnoldi with purification for the shift-invert transformation, Math. Comput. 66 (218) (1997) 667–689.

[11] A. Ruhe, Rational Krylov sequence methods for eigenvalue computation, Linear Algebra Appl. 58 (1984) 391–405.

[12] A. Ruhe, Rational Krylov algorithms for nonsymmetric eigenvalue problems, in: G. Golub, A. Greenbaum, M. Luskin (Eds.), Recent Advances in Iterative Methods, IMA Volumes in Mathematics and its Applications 60, Springer-Verlag, New York, 1994, pp. 149–164.

[13] A. Ruhe, Rational Krylov algorithms for nonsymmetric eigenvalue problems, II: matrix pairs, Linear Algebra Appl. 197-198 (1994) 283–296.

[14] A. Ruhe, The rational Krylov algorithm for nonsymmetric eigenvalue problems. III: complex shifts for real matrices, BIT 34 (1994) 165–176.

[15] A. Ruhe, Rational Krylov: A practical algorithm for large sparse nonsymmetric matrix pencils, SIAM J. Sci. Comput. 19 (5) (1998) 1535–1551.

[16] V. Druskin, L. Knizhnerman, Extended Krylov subspaces: approximation of the matrix square root and related functions, SIAM J. Matrix Anal. Appl. 19 (3) (1998) 755–771.

[17] D. C. Sorensen, Implicit application of polynomial filters in a k-step Arnoldi method, SIAM J. Matrix Anal. Appl. 13 (1) (1992) 357–385.

[18] R. B. Morgan, On restarting the Arnoldi method for large nonsymmetric eigenvalue problems, Math. Comput. 65 (215) (1996) 1213–1231.

[19] R. B. Lehoucq, D. C. Sorensen, Deflation techniques for an implicitly restarted Arnoldi iteration, SIAM J. Matrix Anal. Appl. 17 (4) (1996) 789–821.

[20] G. Stewart, A Krylov-Schur algorithm for large eigenproblems, SIAM J. Matrix Anal. Appl. 23 (3) (2001) 601–614.

[21] G. De Samblanx, K. Meerbergen, A. Bultheel, The implicit application of a rational filter in the RKS method, BIT 37 (4) (1997) 925–947.

[22] M. Berljafa, S. Güttel, Generalized rational Krylov decompositions with an application to rational approximation, SIAM J. Matrix Anal. Appl. 36 (2) (2015) 894–916.

[23] R. Vandebril, Chasing bulges or rotations? A metamorphosis of the QR-algorithm, SIAM J. Matrix Anal. Appl. 32 (1) (2011) 217–247.

[24] R. Vandebril, D. S. Watkins, A generalization of the multishift QR algorithm, SIAM J. Matrix Anal. Appl. 33 (3) (2012) 759–779.

[25] R. Vandebril, D. S. Watkins, An extension of the QZ algorithm beyond the Hessenberg-upper triangular pencil, Electron. Trans. Numer. Anal. 40 (2013) 17–35.

[26] D. Camps, K. Meerbergen, R. Vandebril, A rational QZ method`arXiv:1802.04094`.

[27] T. Mach, R. Vandebril, On deflations in extended QR algorithms, SIAM J. Matrix Anal. Appl. 35 (2) (2014) 559–579.

[28] C. Lanczos, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, J. Res. Natl. Bur. Stand. 45 (4) (1950) 255.

[29] M. Berljafa, Rational Krylov Decompositions: Theory and Applications, Ph.D. thesis, The University of Manchester (2017).

[30] R. Vandebril, M. Van Barel, N. Mastronardi, A parallel QR-factorization/solver of quasiseparable matrices, Electron. Trans. Numer. Anal. 30 (2008) 144–167.

[31] D. Fasino, Rational Krylov matrices and QR steps on Hermitian diagonal-plus-semiseparable matrices, Numer. Linear Algebr. with Appl. 12 (8) (2005) 743–754.

[32] M. Van Barel, D. Fasino, L. Gemignani, N. Mastronardi, Orthogonal Rational Functions and Structured Matrices, SIAM J. Matrix Anal. Appl. 26 (3) (2005) 810–829.

[33] T. Mach, M. V. Barel, R. Vandebril, Inverse eigenvalue problems for extended hessenberg and extended tridiagonal matrices, Journal of Computational and Applied Mathematics 272 (2014) 377 – 398.

[34] B. Kågström, P. Poromaa, Computing eigenspaces with specified eigenvalues of a regular matrix pair (A, B) and condition estimation: theory, algorithms and software, Numer. Algorithms 12 (1996) 369–407.

[35] P. Van Dooren, A generalized eigenvalue approach for solving Riccati equations, SIAM Journal on Scientific and Statistical Computing 2 (2) (1981) 121–135.

[36] J. Aurentz, T. Mach, L. Robol, R. Vandebril, D. Watkins, Core-Chasing Algorithms for the Eigenvalue Problem, Fundamentals of Algorithms, Society for Industrial and Applied Mathematics, 2018.

[37] R. Van Beeumen, K. Meerbergen, W. Michiels, Connections between contour integration and rational Krylov methods for eigenvalue problems, Department of Computer Science, KU Leuven, Technical report TW 673 (2016).

[38] J. Baglama, D. Calvetti, L. Reichel, IRBL: an implicitly restarted block-Lanczos method for large-scale Hermitian eigenproblems, SIAM J. Sci. Comput. 24 (5) (2003) 1650–1677.

[39] J. Baglama, L. Reichel, An implicitly restarted block Lanczos bidiagonalization method using Leja shifts, BIT 53 (2) (2013) 285–310.

[40] J. S. Turner, Buoyancy Effects in Fluids, Cambridge University Press, 1973.

[41] K. A. Cliffe, T. J. Garratt, A. Spence, Eigenvalues of the discretized Navier-Stokes equation with application to the detection of Hopf bifurcations, Adv. Comput. Math. 1 (3) (1993) 337–356.

[42] H. Elman, K. Meerbergen, A. Spence, M. Wu, Lyapunov inverse iteration for identifying Hopf bifurcations in models of incompressible flow., SIAM J. Sci. Comput. 34 (3) (2012) 1584–1606.

[43] H. C. Elman, A. Ramage, D. J. Silvester, IFISS: A computational laboratory for investigating incompressible flow problems, SIAM Rev. 56 (2) (2014) 261–273.

[44] H. C. Elman, M. Wu, Lyapunov inverse iteration for computing a rightmost eigenvalues of large generalized eigenvalue problems, SIAM J. Matrix Anal. Appl. 34 (4) (2013) 1685–1707.

[45] R. B. Lehoucq, D. C. Sorensen, C. Yang, ARPACK Users' Guide: Solution of Large Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods., 1997.