Using implicitly filtered RKS for generalised eigenvalue problems

Gorik De Samblanx, Adhemar Bultheel *

Department of Computer Science, K.U.Leuven, Celestijnenlaan, 200 A, B-3001 Leuven, Belgium

Received 3 October 1997; accepted 25 February 1999

Abstract

The rational Krylov sequence (RKS) method can be seen as a generalisation of Arnoldi’s method. It projects a matrix pencil onto a smaller subspace; this projection results in a small upper Hessenberg pencil. As for the Arnoldi method, RKS can be restarted implicitly, using the QR decomposition of a Hessenberg matrix. This restart comes with a projection of the subspace using a rational function. In this paper, it is shown how the restart can be worked out in practice. In a second part, it is shown when the filtering of the subspace basis can fail and how this failure can be handled by deflating a converged eigenvector from the subspace, using a Schur-decomposition.

MSC: 65F15

Keywords: Rational Krylov method; Implicitly restarted Arnoldi; Generalised eigenvalue problem; Shift-invert

1. Introduction

The rational Krylov sequence (RKS) algorithm [11-14] is an algorithm that finds a limited set of eigenvalues of a generalised eigenvalue problem

\[ Ax = \lambda Bx, \]

in a certain region of the complex plane. Suppose that \( A \) has full rank. Eigenvalues in the focussed region are called ‘wanted’ eigenvalues, they can be rightmost eigenvalues, the largest or the smallest eigenvalues, … RKS extends the idea of shift-invert Arnoldi [9,15,16] by computing a rational Krylov subspace

\[ \text{span}\{v_1, v_2 = S_1w_1, v_3 = S_2w_2, \ldots\}, \text{ with } S_i = (A - \mu_i B)^{-1} B \text{ and } w_i \in \{v_1, \ldots, v_i\} \]

* Corresponding author. Tel.: 32-16-327540; fax: 32-16-327996.
E-mail address: adhemar.bultheel@cs.kuleuven.ac.be (A. Bultheel)

0377-0427/99 - see front matter © 1999 Elsevier Science B.V. All rights reserved.
PII: S0377-0427(99)00089-8
instead of a regular Krylov subspace with a fixed $\mu$. The subspace is spanned by an orthogonal basis $V_k$. The eigenvalue problem is then projected onto this subspace. From the projected problem a sequence of approximate eigenvalues and eigenvectors is computed that, under mild assumptions, converges to a certain eigenpair.

However, the size of the subspace can become too large. E.g., if more than one eigenvalue is sought, then it is not necessary to extend $V_k$ each time with a large number of vectors. When an eigenvalue is found and a new, different eigenvalue is wanted, $V_k$ contains a lot of superfluous information for the computation of that eigenvalue. On the other hand, it is a waste of computational effort to restart the algorithm totally and to begin with a completely new subspace. For Arnoldi's method, the Implicitly Restarted Arnoldi method [4,17] was proposed as a solution to this problem. Similarly, the RKS method can be restarted implicitly [2] or explicitly. The implicit restart can be done without losing too much relevant information. Implicitly restarting the RKS method then corresponds to filtering the subspace with a rational filter

$$V_{k-p} \rightarrow \prod_{i=0}^{\mu-1} (A - \mu_iB)^{-1}(\alpha_iA - \beta_iB)V_{k-p},$$

where the $\alpha_i$, $\beta_i$ may be chosen freely. Therefore, it was called Implicitly Filtered RKS (IFRKS).

In [7], it was noticed that the filtering property can be used to filter away spurious eigenvalues. Indeed, if the matrix $B$ is singular, then the problem has an infinite eigenvalue (possibly defective) that will be approximated by large finite eigenvalues. These spurious eigenvalues can mislead the algorithm, e.g. when the largest finite eigenvalues are needed. They can also have an important influence on the accuracy of the solution. The eigenvectors of the infinite eigenvalue lie in the null-space of the columns of $B$. Restarting the subspace with $\alpha_i = 0$, will then remove these spurious eigendirections from the basis $V_k$.

In this text, we show how the restarting algorithm that has been presented in [2], can be used safely to shrink the subspace $V_k$. Hereby, the two classical sources of numerical errors must be considered: the implementation must be stable, but the problem can be ill-conditioned anyway. The computation of the matrices that are involved with IFRKS can be done in different ways. It turns out that the choice of the algorithm becomes important in the case where the implicit restart itself ‘fails’. If IFRKS is applied to a near-singular matrix, then the filtering property (1) can be lost. This is the case when the algorithm is used to remove a converged eigenvector. This error does not depend on how the restart matrices are computed. As a solution to this problem, we show in the last part of the text an alternative way to restart RKS by truncating the RKS relation. Truncation can be used to deflate converged eigenvectors or, more generally, to restart the RKS algorithm in one big step. However, no implicit filtering can be applied while doing so.

The text is structured as follows. In Section 2, we review the RKS algorithm and we show how the approximate eigenvalues can be computed from the projected system. Section 3 derives different ways to use IFRKS in order to compute a restarted RKS relation. Section 4 focuses on the errors that are involved with restarting. It is shown there that IFRKS may be inaccurate if a converged relation is restarted. A solution to this problem is handled in Section 5, where is shown how Ritz vectors can be deflated from the relation. The proposed method is able to deflate a vector that corresponds to any Ritz or Harmonic Ritz value. In Section 6, we give an example of the use of an algorithm that combines these methods. Section 7 closes the text with some conclusions.
Notation 1.1. Matrices are denoted by upper case roman characters. The index of a matrix is equal to its number of columns this is also the iteration step in which it is constructed). The $k \times k$ leading submatrices of the (rectangular) $K_k$, $H_k$, $T_k$ are denoted by $K_k$, $H_k$, $T_k$. The range of the columns of a matrix $V$ is denoted by $\mathcal{R}(V)$. The $i,j$th element of a matrix $H$ is denoted by $(H)_{i,j}$. Lower case roman characters are vectors and scalars are denoted by Greek characters, $\bar{\alpha}$ denotes the complex conjugate of a scalar. $X^*$ denotes the Hermitian transpose and $\| \cdot \|$ denotes the 2-norm, whereas $\| \cdot \|_F$ stands for the Frobenius norm. We call $V$ an orthogonal matrix if $V^* V = I$, and $V$ is called unitary when it is also square. The machine precision is denoted by $u$.

2. The RKS algorithm and RKS triples

In this section, we recall the RKS algorithm. RKS stores its eigenvalue information in a set of matrices that we call RKS triples. An RKS triple is defined such that it always corresponds to a run of an RKS algorithm, although it might have been computed in a different manner.

The RKS algorithm is presented in Algorithm 1. It defines the notation for some important matrices that are involved in the remainder of the text.

Algorithm 1. RKS

Input: $v_1 \in \mathbb{C}^n$, $\| v_1 \| = 1$.

0. Let $V_1 = [v_1]$.
1. For $i = 1, \ldots, k$ do
   1.1. Select a pole $\mu_i$ and a continuation vector $t_i \neq 0 \in \mathbb{C}^i$.
   1.2. Form $w_i = (A - \mu_i B)^{-1} B V_i t_i$.
   1.3. Orthogonalise $w_i$ against the columns of $V_i$ and let $h_i = V_i^* w_i$.
   1.4. Normalise $v_{i+1} = w_i/\eta_i$, with $\eta_i = \| w_i \|$.
   1.5. Compute the approximate eigenpair $(\theta_i, y_i)$.
   1.6. Update $V_{i+1} = [V_i v_{i+1}]$.

In each sweep, Algorithm 1 computes a vector $w_i = (A - \mu_i B)^{-1} B V_i t_i$, which is orthogonalised and added to the basis $V_i$

$$V_{i+1} = [V_i v_{i+1}] = [V_i (w_i - V_i h_i)/\eta_i],$$

with

$$h_i = V_i^* w_i \quad \text{and} \quad \eta_i = \| w_i - V_i h_i \|.$$ 

The scalar $\mu_i$ is called the pole, $t_i \in \mathbb{C}^i$ is the continuation vector. It is easy to see that

$$AV_{i+1} \begin{bmatrix} h_i \\ \eta_i \end{bmatrix} = BV_{i+1} \begin{bmatrix} \mu_i & h_i \\ \eta_i & 0 \end{bmatrix} + \begin{bmatrix} t_i \\ 0 \end{bmatrix}.$$ 

Summarising this information for the first $k$ steps, we get the RKS relation

$$AV_{k+1} H_k = B V_{k+1} K_k,$$  (2)
where \( K_k, H_k \in \mathbb{C}^{k+1 \times k} \) are unreduced upper Hessenberg matrices and \( V_{k+1}^* V_{k+1} = I \). (An unreduced upper Hessenberg matrix is a matrix that has nonzero subdiagonal elements.) If we collect the poles in a diagonal matrix \( M_k = \text{diag}(\mu_i) \in \mathbb{C}^{k \times k} \) and the continuation vectors in an upper triangular matrix \( T_k \in \mathbb{C}^{k+1 \times k} \), then
\[
K_k = H_k M_k + T_k.
\]
(3)

Any pair of unreduced Hessenberg matrices \( H_k \) and \( K_k \) can be decomposed in a unique way as in (3). Therefore, any relation (2) with unreduced Hessenberg matrices corresponds uniquely to an RKS process with a certain starting vector, a set of poles
\[
\mu_i \equiv (K_k)_{i+1,i}/(H_k)_{i+1,i}, \quad i = 1, \ldots, k,
\]
and continuation vectors, assuming that none of the poles is an eigenvalue of \((A,B)\). The set of matrices \((V_k, H_k, K_k)\) that fulfills the RKS relation, is then called an RKS triple.

**Definition 2.1.** The set of matrices \((V_k, H_k, K_k)\), with \( V_k \in \mathbb{C}^{n \times k+1} \) orthogonal and \( H_k, K_k \in \mathbb{C}^{k+1 \times k} \) upper Hessenberg is called an RKS triple of order \( k \) for \((A,B)\) if
(i) they fulfill (2),
(ii) the matrix \( H_k \) is unreduced,
(iii) none of the \( \mu_i \), defined as in (4), is an eigenvalue of \((A,B)\).

If \( H_k \) would be reduced, then \( V_k \) would contain an invariant subspace for \((A,B)\). Therefore, we assume in this text that \( \mathcal{R}(V_k) \) does not contain an exact eigenvector of \((A,B)\).

It should be noted that (2) can be shifted. Consequently, every restarting procedure that is used on a certain RKS triple, can also be applied on its shifted companion. This leads to a more flexible use of the algorithm. The following lemma shows how an RKS relation and its RKS triple are shifted. The extra term \( S_k \) can be seen as an error term on the RKS relation. It will be used in Section 4.

**Lemma 2.2.** Suppose that \( AV_{k+1} = BV_{k+1} K_k + S_k \). Given a set of scalars \( \tau, \alpha, \beta \), then this relation can be rewritten as
\[
(A - \tau B)V_{k+1}(\alpha K_k - \beta H_k) = (\alpha A - \beta B)V_{k+1}(K_k - \tau H_k) + (\tau \alpha - \beta) S_k.
\]
(5)

Suppose that \( S_k \equiv 0 \) and that \((V_{k+1}, H_k, K_k)\) is an RKS triple of order \( k \) for \((A,B)\). If \( \alpha \mu_i \neq \beta \), for \( i = 1, \ldots, k \), and if \( \tau \) is no eigenvalue of \((A,B)\), then \((V_{k+1}, \alpha K_k - \beta H_k, K_k - \tau H_k)\) is an RKS triple of order \( k \) for \((A - \tau B, \alpha A - \beta B)\).

**Proof.** The first part of the proof is obvious. The second part corresponds to Lemma 3.3 in [2].

2.1. *Computation of the approximate eigenvalues*

The small pencil \((K_k, H_k)\) can be viewed as a projection of the large matrix pencil \((A,B)\). However, the approximate eigenvalues can not be computed directly from these matrices, since they are rectangular. Therefore, we must reduce \((K_k, H_k)\) to a (generalised) eigenvalue problem. There are different ways to do so.
Definition 2.3 (Morgan [8], Paige et al. [10]). Given a matrix $A$ and a subspace basis $V_k$. Then $(\theta, y = V_kz)$ is called a Ritz pair of $A$ (in $\mathcal{A}(V_k)$) with respect to $\mathcal{R}(V_k)$ if

$$Ay - \theta y \perp V_k,$$

$(\theta, y)$ is a Harmonic Ritz pair of $A$ with respect to $\mathcal{R}(V_k)$ if

$$Ay - \theta y \perp AV_k.$$

Applying this definition on the reduced eigenvalue problem of the RKS relation, gives the following result.

Lemma 2.4. Given an RKS triple $(V_{k+1}, H_k, K_k)$ of order $k$ for $(A, B)$. Suppose that $B$ is nonsingular. Then $(\theta, y = V_{k+1}H_kz)$ is a Ritz pair of $B^{-1}A$ with respect to $\mathcal{R}(V_{k+1}H_k)$ if

$$H_k^*K_kz - \theta H_k^*H_kz = 0.$$

$(\theta, y = V_{k+1}H_kz)$ is a Harmonic Ritz pair of $B^{-1}A$ with respect to $\mathcal{R}(V_{k+1}H_k)$ if

$$K_k^*K_kz - \theta K_k^*H_kz = 0.$$

Given a scalar $\delta \in \mathbb{C}$. The solution $(\theta - \delta, y = V_{k+1}H_kz)$ of

$$(K_k - \delta H_k)^*K_kz - \delta(K_k - \delta H_k)^*H_kz = 0$$

is a Harmonic Ritz pair of $B^{-1}(A - \delta B)$.

Proof. The first part is proven in [5]. The second part can be shown as follows. From the Definition 2.2 of a Harmonic Ritz pair of $B^{-1}A$, we have

$$B^{-1}AV_{k+1}H_kz - \theta V_{k+1}H_kz \perp B^{-1}AV_{k+1}H_k.$$

By Eq. (2), this is equivalent to

$$V_{k+1}K_kz - \theta V_{k+1}H_kz \perp V_{k+1}K_k$$

and thus, using $V_{k+1}^*V_{k+1} = I$,

$$K_k^*K_kz - \theta K_k^*H_kz = 0.$$

The last result can be found by shifting (2) as in Lemma 2.1, with $\alpha = 1$, $\beta = -1$ and $\tau = \delta$. In the former, $K_k$ must then be replaced by $K_k - \delta H_k$ and $A$ by $A - \delta B$. Hence,

$$(K_k - \delta H_k)^*(K_k - \delta H_k)z - (\theta - \delta)(K_k - \delta H_k)^*H_kz = (K_k - \delta H_k)^*K_kz - \theta(K_k - \delta H_k)^*H_kz = 0.$$

In the remainder of this text, we suppose that there always exist $k$ linear independent eigenvectors $z_i$ for the eigenvalue problem (6), i.e.

$$\text{rank}([z_1 \cdots z_k]) = k \quad \text{and} \quad \mathcal{R}(V_{k+1}H_k) = \mathcal{R}(V_{k+1}H_k[z_1 \cdots z_k]) = \mathcal{R}([y_1 \cdots y_k]).$$
A special case is found by choosing $\delta = \mu_k$. Since the last row of $K_k - \mu_k H_k$ is zero, the corresponding Harmonic Ritz values are found by solving

$$(K_k - \mu_k H_k)^*(K_k - H_k)z = (K_k - \mu_k H_k)^*(K_k - H_k)z = 0,$$

which clearly corresponds to

$$K_k z - \mu_k H_k z = 0.$$  \hfill (8)

If the wanted eigenvalues lie in the interior of the spectrum of $(A, B)$, then using Harmonic Ritz values can turn out to be advantageous. However, the importance of the difference between ‘regular’ Ritz values and Harmonic Ritz values for nonsymmetric problems is not clear at this moment. In the remainder of this text, we will use the last option (8) to compute the approximate eigenpair, since it seems to be a most natural choice for RKS.

We will measure the level of convergence of the algorithm by the residual norm of the approximate eigenvector

$$\| r_k \| = \| A y_k - \theta_k B y_k \| = \| AV_{k+1} H_k z_k - \theta_k B V_{k+1} H_k z_k \|,$$

which corresponds by Eqs. (2) and (8) to

$$\| r_k \| = |\eta_k z_{k,k}||\mu_k - \theta_k||Bv_{k+1}|,$$

where $z_{k,k}$ is the $k$th component of $z_k$. We assume that $\|Bv_{k+1}\| = O(1)$. The residual norm will be small if $|\eta_k z_{k,k}|$ and/or $|\mu_k - \theta_k|$ is small. If we do not change the pole $\mu_k$ in every step, then $|\mu_k - \theta_k|$ will be larger than $|\eta_k z_{k,k}|$. However, it is not necessary for $|\eta_k|$ to be small when the method converges to some eigenvector. But if the method converges, then $|\eta_k z_{k,k}|$, which is a measure for the contribution of $v_{k+1}$ in $y_k$, must clearly tend to zero.

3. Applying IFRKS

The RKS subspace $R(V_k)$ can be filtered with an implicit restart, using the information in the matrices $K_k$ and $H_k$. We derive in this section different ways to compute this restart. In the next section, we will show that even when things are carefully computed, the filtering of $V_k$ can fail.

First, we recall how the RKS relation can be restarted.

**Theorem 3.1.** Given $(V_{k+1}, H_k, K_k)$, an RKS triple of order $k$ for $(A, B)$. Given a nontrivial set of scalars $(\alpha, \beta)$, with $\alpha \mu_i \neq \beta$, $i = 1, \ldots, k$. Say that $Q$ is the orthogonal matrix that is computed from the $QR$ factorisation

$$\alpha K_k - \beta H_k = [Q \ R] \begin{bmatrix} R \ 0 \end{bmatrix}, \quad Q \in \mathbb{C}^{k+1 \times k}, \ R \in \mathbb{C}^{k \times k},$$

and $R$ is an upper triangular matrix. If $Z \in \mathbb{C}^{k \times k-1}$ is an upper Hessenberg matrix such that

$$q^* H_k Z = 0 \ \text{and} \ q^* K_k Z = 0,$$  \hfill (11)
then $Z$ must be unreduced and $(V^+_k, H^+_{k-1}, K^+_{k-1})$, defined by

$$V^+_k \equiv V_{k+1}^*, \quad H^+_{k-1} \equiv Q^*H_kZ \quad \text{and} \quad K^+_{k-1} \equiv Q^*K_kZ,$$

is an RKS triple of order $k-1$ for $(A,B)$. Moreover,

$$\mathcal{R}(V^+_k) = \mathcal{R}((A - \mu_k B)^{-1}(zA - \beta B)V_k).$$

If the shifts are chosen as $(z, \beta) = (1, \theta_k)$, with $\theta_k$ the Ritz value corresponding to a Ritz vector $y_k$, then the Ritz values of the new RKS triple are the same as the old ones, except for $\theta^+_1 = \theta_1, \ldots, \theta^+_{k-1} = \theta_{k-1}$.

**Proof.** The theorem corresponds to Theorem 4.1 and Theorem 5.2 in [2].

**Computation of $Q$.** The computation of $Q$ is a straightforward result of Theorem 3.1. Once we have set the parameters $\alpha$ and $\beta$, then $Q$ is the orthogonal, unreduced upper Hessenberg matrix in the (skinny) QR factorisation (10). The computation of $Q$ is cheap, considered that $Q$ has to be orthogonalised anyway, in order to get an orthogonal $V^+_k$. However, the QR decomposition is a process that is only backward stable, so forward errors may be introduced. In the next section, we will focus further on this problem.

**Computation of $Z$.** In contrast to the computation of $Q$, there are many degrees of freedom left for the computation of $Z$. Indeed, any full rank upper triangular matrix $U \in \mathbb{C}^{k-1 \times k-1}$ corresponds to a new matrix $Z^U \equiv ZU$ that fulfils the conditions of Theorem 3.1. $U$ does not change the filtering properties of the restart, nor the new approximate eigenvalues, since it only replaces $(K^+_{k-1}, H^+_{k-1})$ by $(K^+_{k-1}U, H^+_{k-1}U)$. Therefore, we will look for a $Z$ and a $U$ that are cheap to compute and that make the restarting procedure as robust as possible. If we multiply (3) with $U$ before the restart, then we get

$$K^U_k = K_kU = H_kUM_k + (T_kU + H_kM_kU - H_kUM_k) = H^U_kM_k + T^U_k,$$

with $T^U_k = K_kU - H_kUM_k$.

**Lemma 3.2.** Given the orthogonal matrix $Q \in \mathbb{C}^{k+1 \times k}$ that fulfils (10) and say that $\bar{Q} \in \mathbb{C}^{k \times k-1}$ is the upper left submatrix of $Q$. Suppose that $T_k$ and $zM_k - \beta I$ have full rank. Then $Z = (zM_k - \beta I)(T_k)^{-1}\bar{Q}$ fulfils condition (11) and it leads to upper Hessenberg matrices $K^+_{k-1}$ and $H^+_{k-1}$, where the latter is unreduced.

**Proof.** First note that, because $Q$ is upper Hessenberg, $q^*Q = 0$ implies $q^*\begin{bmatrix} \bar{Q} \\ 0 \end{bmatrix} = q^*I_{k+1,k}\bar{Q} = 0$. If

$$QR = (zK_k - \beta H_k) = H_k(zM_k - \beta I) + zT_k,$$

then $q^*(zK_k - \beta H_k) = 0$. Combining this with the definition of $Z$ gives

$$q^*H_kZ = q^*H_k(zM_k - \beta I)(T_k)^{-1}\bar{Q} = q^*(zK_k - \beta H_k)(T_k)^{-1}\bar{Q} - zq^*T_k(T_k)^{-1}\bar{Q} = 0 + 0,$$
since \(q^* T_k (T_k)^{-1} \tilde{Q} = q^* I_{k+1,k} \tilde{Q} = 0\). On the other hand, \(q^* K_k Z = \frac{1}{2} q^*(zK_k - \beta H_k)Z + \frac{\bar{\eta}}{2} q^* H_k Z = 0\), so both conditions in Eq. (11) are fulfilled (if \(\eta = 0\), then \(K_k\) and \(H_k\) must be swapped). By using Eq. (15), we get

\[
H_{k-1}^+ = Q^* H_k Z = R (T_k)^{-1} \tilde{Q} - z I_{k,k-1},
\]

\[
K_{k-1}^+ = Q^* K_k Z = R M_k (T_k)^{-1} \tilde{Q} - \beta I_{k,k-1},
\]

so these matrices are equal to the product of a full rank upper triangular matrix and an unreduced upper Hessenberg matrix, plus a diagonal matrix. Therefore, \(K_{k-1}^+\) and \(H_{k-1}^+\) must be unreduced upper Hessenberg.

The choice of \(Z\) that is proposed in Lemma 3.2 is easy to apply. It is also not very expensive to implement. However, it heavily depends on the inversion of the matrix \(T_k\). If \(T_k\) is nearly singular, then the restarting procedure can be unstable. This is often the case when the continuation vector is chosen \(\eta_k \equiv z_k\): since we hope that the method will converge, we can expect that \(z_k \simeq z_{k+1}\). But even if \(T_k\) would be the unit matrix, then after a few steps of a repeated restart, the matrix can get a small singular value. However, there is a solution to this problem. We can multiply Eq. (2) by an upper triangular matrix \(U\) before restarting. If \(U\) can be chosen such that \(T_k^U\) is e.g. a diagonal matrix, then \(Z\) can be computed in a more stable way. The next lemma shows that such a \(U\) can always be found.

**Lemma 3.3.** Consider an RKS relation (2). Given \(F = \{ T_k^U \mid T_k^U = K_k U - H_k U M_k, \ U \text{ is upper triangular with } \text{diag}(U) = 1 \}\). Then there exist at least one \(T_k^U \in F\) that is a diagonal matrix.

**Proof.** The \(i\)th column \(u_i\) of \(U\) must fulfil

\[
\eta_i e_i = (K_i - \mu_i H_i) u_i \quad (e_i \text{ is the } i\text{th unit vector}),
\]

for some \(\eta_i \neq 0, i = 1, \ldots, k\). This is always possible since

\[
K_i - \mu_i H_i = \begin{bmatrix} K_i - \mu_i H_i & 0 \\ \vdots & \ddots & \ddots \\ 0 & \cdots & 0 & K_i - \mu_i H_i \end{bmatrix}
\]

has full rank ([2], Lemma 3.2(B)).

Following Lemma 3.3, there exists always an equivalent RKS relation that can be filtered in a stable way. The computation of a suitable matrix \(U\) does not have to be as expensive as it seems. Indeed, often a pole \(\mu_i\) is kept constant for several steps, and we can restrict ourselves to the columns of \(T_k\) that have a small diagonal element. But there is no guarantee that none of the \(K_i - \mu_i H_i\) is nearly singular or that the multiplication with \(U\) itself is well-conditioned. Therefore, we will use \(U\) only to balance the norms of the columns of \(zK_k - \beta H_k\).

**Computing \(Z\) without inverting \(T\).** We can avoid the inversion of the matrix \(T_k\), without computing the \(U\). If we set

\[
Z_k^i = \tilde{Q}^* T_k (z M_k - \beta I)^{-1} \in \mathbb{C}^{k-1 \times k},
\]

(16)
then we need to find an upper Hessenberg matrix $Z$ such that $Z^\dagger Z = I$. Since we only must know $Z$ up to multiplication with some upper triangular matrix $U$, as in (14), it is sufficient to compute a $Z^\perp$ such that $Z^\dagger Z^\perp = U$. If we set $g^*$ equal to the last row of $Z^\dagger$, i.e., $g^* = e_k^* Z^\dagger$, then an appropriate $Z^\perp$ fulfills

$$g^* Z^\perp = [0 \cdots 0 1].$$

By Lemma 3.4, there exists such a $Z^\perp$ that it is upper Hessenberg. The same lemma states that, since $Z^\perp = ((z M_k - \beta I) T_k^{-1} \hat{Q}) U$, it is unreduced.

**Lemma 3.4.** Given a nonzero vector $g \in \mathbb{C}^k$, there exists at least one orthogonal upper Hessenberg matrix $Z \in \mathbb{C}^{k \times k-1}$ such that $g^* Z = 0$. If the first entry of $g$ is nonzero, i.e., $g^* e_1 \neq 0$, then $Z$ is unreduced. If $g \neq 0$, then for any other full rank upper Hessenberg matrix $\hat{Z}$ that fulfills $g^* \hat{Z} = 0$, there exists a full rank upper triangular matrix $U \in \mathbb{C}^{k \times k}$ such that $\hat{Z} = ZU$.

**Proof.** Suppose that the first $k - l - 1 \geq 0$ elements of $g$ are zero. We can write $g^* = [0 \, \gamma \, g^{(l)}]$, with $0 \neq \gamma \in \mathbb{C}$ and $g^{(l)} \in \mathbb{C}^{1 \times l}$, then an appropriate $Z$ is given by

$$Z = \begin{bmatrix}
I_{k-l-1} & 0 \\
0 & -g^{(l)} \\
0 & \gamma \end{bmatrix} S,$$

where $S \in \mathbb{C}^{k \times k}$ is an upper triangular matrix such that $Z^* Z = I_{k-1}$. If $g^* e_1 \neq 0$, then $l = k - 1$ and $Z$ is unreduced. If there exists a full rank Hessenberg matrix $\hat{Z}$ such that $g^* \hat{Z} = 0$, then $Z$ and $\hat{Z}$ have identical null-spaces, i.e., $\mathcal{R}(g)$. Hence, $\mathcal{R}(Z) = \mathcal{R}(\hat{Z})$, so there exists a full rank matrix $U$ such that $\hat{Z} = ZU$. Note that, since both $Z$ and $\hat{Z}$ are upper Hessenberg, $U$ must be upper triangular. Since $(\hat{Z})_{i+1,i} = (Z)_{i+1,i}(U)_{i,i}$, a subdiagonal element of $\hat{Z}$ is zero if and only if the same subdiagonal element of $Z$ is zero. 

**Algorithm 2 (De Samblanx et al. [2]).** Compute $Z$, such that $g^* Z = 0$.

**Input:** $g \equiv [0 \, \ldots \, 0 \, g_i, g_{i+1} \, \ldots \, g_k] \in \mathbb{C}^k$, with $g_i \neq 0$, $l \leq 1$.

**Output:** $Z \in \mathbb{C}^{k \times k-1}$ orthogonal upper Hessenberg, with $g^* Z = 0$.

1. Set $\hat{Z}_i = I_{k-l-1}, k-1$.
2. For $i = 1, \ldots, k - 1$
   1. Find $m$ such that $|g_m| = \max \{|g_j|, j \leq i\}$.
   2. Set $\hat{Z}_{i+1} = \begin{bmatrix} \hat{Z}_i & -g_{i+1} \end{bmatrix}$.
3. Compute $S$ such that with $Z = \hat{Z}_k S$, $Z^* Z = I$.

Algorithm 2 computes the matrix $Z$. The approach of this algorithm differs slightly from Lemma 3.4: instead of choosing $\gamma \equiv g_i$ as the subdiagonal element (before the orthogonalisation), we choose the largest element $g_m$ in $\{g_i, \ldots, g_k\}$. This might improve the numerical properties of the algorithm in cases where $\gamma$ is very small, but nonzero.

**Computing directly an orthogonal $Z$.** There is a third possible strategy to compute a suitable $Z$. We can construct $Z^\perp$ explicitly such that it fulfills (11) and such that it is orthogonal. Since by
construction, \( q^*(zK_k - \beta H_k) = 0 \), it is sufficient to find a matrix \( Z^\perp \) that zeros out any other linear combination of \( q^*K_k \) and \( q^*H_k \).

**Lemma 3.5.** Given the vector \( q \) as defined in Eq. (10) and say that \( Z^\perp \) is an orthogonal upper Hessenberg matrix such that
\[
q^*(\tilde{\beta}K_k + \tilde{\alpha}H_k)Z^\perp = 0,
\]
then \( Z^\perp \) fulfills condition (11) and it leads to upper Hessenberg matrices \( K_{k-1}^+ \) and \( H_{k-1}^+ \), where the latter is unreduced.

**Proof.** If \( Z^\perp \) fulfills Eq. (17) then we can derive, reminding that \( q^*(zK_k - \beta H_k) = 0 \),
\[
q^*K_kZ^\perp = \frac{1}{|\alpha|^2 + |\beta|^2} (\tilde{\alpha}q^*(zK_k - \beta H_k) + \beta q^*(\tilde{\beta}K_k + \tilde{\alpha}H_k)) Z^\perp = 0.
\]
The result for \( q^*H_kZ^\perp = 0 \) is analogous. Again, by Lemma 3.4, such a \( Z^\perp \) must exist. Since it has the same nullspace as \( (zM_k - \beta I)\tilde{L}_k^{-1} \tilde{Q} \), which is unreduced, it must be unreduced as well. □

It is easy to see that, upon normalisation of its columns, the orthogonal matrix \( Z^\perp \) is unique. Although this approach seems rather ad hoc, it turns out to be in many cases a very good choice for \( Z \).

**Example 3.1.** Let us illustrate these results with a simple example. We constructed a rather trivial \( 100 \times 100 \) matrix \( A \) that has eigenvalues \( \lambda_i = (i - 1)/100 \), for \( i = 1, \ldots, 100 \):
\[
A = \begin{bmatrix}
99 & 1/100 & 0 & \cdots & 0 \\
0 & 98 & 0 & \cdots & 0 \\
0 & 0 & \ddots & \ddots & 0 \\
0 & \cdots & 0 & 1/100 & 1/100 \\
0 & \cdots & 0 & 0 & 0 \\
\end{bmatrix}
\]
We iterated 8 times with a fixed shift \( \mu_{1\ldots8} = 1.5 \), and a random starting vector, in order to ‘locate’ the rightmost eigenvalue (i.e. \( \lambda = 0.99 \)). Then we iterated a few times with the shift equal to the best approximation of this eigenvalue, i.e. with \( \mu_i = \theta_{\text{rightmost}} \), until convergence.

Three different computations of \( Z \) were tested. First, we computed \( Z^{-1} \equiv (zK_k - \beta H_k)\tilde{L}_k^{-1} \tilde{Q} \), where \( N \) is a diagonal matrix that normalises the columns of \( Z^{-1} \) to unit norm. Secondly, we computed \( Z^L \) such that, with \( \tilde{\alpha} \) from Eq. (16), the product \( Z^LZ^L \) is upper triangular. Again, the columns of \( Z^L \) are normalised to unit norm. The last option computes \( Z^\perp \) directly from \( q^*(\tilde{\beta}K_k + \tilde{\alpha}H_k)Z^\perp = 0 \), with \( Z^\perp \) orthogonal, using Algorithm 2. We displayed in Tables 1 and 2 what would happen if we restarted the RKS equation in order to remove two leftmost or two rightmost Ritz values, after step 9 and 10 (the results are generated with Matlab4, the machine precision is \( u = 2.2e^{-16} \)).

The choices of \( Z \) are compared at three different points. We checked whether they influence the filtering property, what happens if a (converged) eigenvalue is removed and whether the RKS relation still holds to high accuracy.
the errors are larger than in Case 1. All the errors for removing a non-converged eigenvalue, e.g. the leftmost one as shown in the first part of Table 2, then the error on the restarted RKS relation (\( k = 9 \)

| \( k \) | Removing two leftmost Ritz values | \( \| A V^+ H^+ - V^+ K^+ \| / \| H^+ \| \) | \( | \theta_1 - \theta_1^* | \) | \( | \theta_2 - \theta_2^* | \) | \( | y_1 - y_1^* \) | \( | (I - \mathcal{P}_{a,b}) V^+ \| \) | \( \kappa_2(Z) \) |
|-------|--------------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| 9 | Removing two leftmost Ritz values | 3.6e - 16 | 3.8e - 16 | 2.5e - 16 | 7.1e - 16 | 1.5e - 16 | 1.2e - 15 | 6.5 | 1.9 | 1.0 | 7.6e - 4 |
| 10 | Removing two leftmost Ritz values | 2.5e - 16 | 2.9e - 16 | 2.5e - 16 | 2.3e - 16 | 4.4e - 16 | 1.1e - 16 | 7.3e - 14 | 3.3e - 14 | 5.7e - 14 | 2.8e - 10 | 9.1e - 11 | 4.4e - 11 | 4.7e - 12 | 6.5 | 2.0 | 1.0 | 8.1e + 8 |
| 10 | Removing two rightmost (converged) Ritz values | 1.8e - 12 | 3.6e - 12 | 4.9e - 16 | 8.7e - 8 | 8.7e - 8 | 5.7e - 8 | 9.7e - 6 | 9.7e - 6 | 7.9e - 6 | 45 | 29 | 1.0 | 8.1e + 8 |

**Table 2**

Removing Ritz values from an RKS relation with IFRKS when one of the eigenvalues has converged. Same legend as Table 1

**Case 1:** Table 1 shows that, as long as no eigenvalue has converged, all three methods are equally accurate. The RKS relation remains correct to computer precision and the approximate eigenvalues and eigenvectors remain unchanged. Also the filtering property is not violated.

**Case 2:** However, Table 2 shows the result of the same operation after an additional step with an ‘optimal’ shift. At that point, the rightmost eigenvalue is approximated up to 13 digits. Since we used \( t_k \equiv H_k z \), the matrix \( \mathcal{T}_k \), that contains the continuation vectors, is almost singular. If we remove a non converged eigenvalue, e.g. the leftmost one as shown in the first part of Table 2, then the errors are larger than in Case 1. All the errors for \( Z^\perp \) are smaller than the errors for the other
choices, but the differences are still minor. The error on the approximated eigenvalues \(|\theta - \theta^+|\) in the table) is smaller than the error on the filtering step (displayed as \(||(I - V_{x,0})V^+||\)). Moreover, the difference between the old and the new Ritz values is smaller that the difference between the Ritz values and the true eigenvalues. So no convergence is lost. Also the error on the RKS relation \(||AV^+H^+ - BV^+K^+ \|/\|H^+\||\) remains small.

Case 3: If we try removing the rightmost Ritz value, which has converged, then the error on the filtering step is much larger, as predicted in Theorem 4.2. This is shown in the second part of Table 2. For the first two choices of \(Z\), the RKS relationship does not hold any more up to computer precision, while it does for the choice of \(Z^\perp\). This can be explained by noticing that only \(Z^\perp\) is generated explicitly in order to make \(q^*H_iZ^\perp = 0 = q^*K_iZ^\perp\), which is the condition for an accurate new RKS relation. For the other two choices, this property is only implicitly fulfilled. This type of loss of accuracy is worse than some loss of convergence, since it can not be recovered: the residual (9) is limited by the error on (2). Therefore, the choice of \(Z^\perp\) is advisable.

We can conclude from the example that the choice of \(Z^\perp\) is preferable, however – at least for this example – the other choices perform better than expected. The main advantage of using \(Z^\perp\) is that it does not introduce errors in the RKS relation. The failure of the filtering with IFRKS has a different cause, which is studied in the next section.

4. Possible errors while using IFRKS

In practice, the computed RKS relation (2) is not exact. In each step of Algorithm 1, a computational error is added to the relation. If the error is small, then it will have no important effect on the convergence properties of the algorithm. However, in [7], it has been shown for the Implicitly Restarted Arnoldi method, that this error can have an important effect on the filtering properties of the restarting algorithm. In this section, we show a similar property for the IFRKS algorithm. We show that, even if the error on the RKS relation is small, the filtering step (13) described in Theorem 3.1 can be inaccurate.

There are two main sources of possible computational errors in Algorithm 1. First, the linear system in step 1.2. of the algorithm will only be solved to some residual error:

\[(A - \mu_iB)w = BV_i t_i + s'_i, \quad \text{with } ||s'_i|| \leq u(||A - \mu_iB|| ||w|| + ||BV_i t_i||),\]

where \(u\) is the machine precision. Secondly, the orthogonalisation steps 1.3. and 1.4. can be inexact:

\[w = V_i h_i + v_{i+1} s''_i.\]

If we call \(s''_i\) the rounding error made on the computation of the \(i\)th column of \(K_i\) and if we sum the errors in the vector \(s_i = s'_i - (A - \mu_iB)s''_i + BV_i s''_i\), then we can collect all the error vectors in one matrix \(S_k = [s_1, \ldots, s_k]\). We get the RKS relationship with rounding errors

\[AV_{k+1}H_k = BV_{k+1}K_k + S_k.\] (18)

In Theorem 4.2, we will show that if the filtering step fails, then the influence of \(S_k\) will generally be larger than the influence of the error on the computed QR decomposition. In order to prove that, we need the following Lemma.
Lemma 4.1. If we call $\sigma_1(T) \geq \sigma_2(T) \geq \cdots$ the singular values of a matrix $T$ and if $R \in \mathbb{C}^{k-1 \times k-1}$ has full rank, then given a vector $r \in \mathbb{C}^{k-1}$ and a scalar $\rho \in \mathbb{C}$,

$$\sigma_{k-1}(\hat{R}) = \sigma_{k-1} \left( \begin{bmatrix} R & r \\ 0 & \rho \end{bmatrix} \right) \leq \kappa \sigma_{k-1}(R) + |\rho|,$$

where $\kappa \leq 1 + \|R^{-1}r\|$.

Proof. By definition,

$$\sigma_i(\hat{R}) = \max_{X \in \mathcal{X}_i} \min_{x \in X} \frac{\|\hat{R}x\|}{\|x\|} = \min_{X \in \mathcal{X}_i} \max_{x \in X} \frac{\|\hat{R}x\|}{\|x\|},$$

where $\mathcal{X}_i \subset \mathbb{C}^k$ is the set of subspaces of dimension $i$. Define $y \in \mathbb{C}^{k-1}$ as the vector with $\|y\| = 1$ such that

$$\|Ry\| = \sigma_{k-1}(R)$$

and set $v = \begin{bmatrix} y \\ 0 \end{bmatrix}$.

If we set

$$u = \begin{bmatrix} R^{-1}r \\ -1 \end{bmatrix},$$

then $[R \, r]u = 0$. Say that the subspace spanned by these vectors is $X_s$, then

$$\mathcal{R}(X_s) = \mathcal{R}(v, u) = \mathcal{R} \left( u, \hat{u} = \frac{1}{N} \begin{bmatrix} R^{-1}r - y(y^*R^{-1}r) \\ -1 \end{bmatrix} \right),$$

where $N = \left\| \begin{bmatrix} R^{-1}r - y(y^*R^{-1}r) \\ -1 \end{bmatrix} \right\| \geq 1$.

Since $[u \, \hat{u}]$ forms an orthogonal basis for $X_s$, any normalised vector in this subspace can be written as $x = \alpha u + \beta \hat{u}$, with $|\alpha|^2 + |\beta|^2 = 1$. Therefore, since $X_s \in \mathcal{X}_2$,

$$\sigma_{k-1}(\hat{R}) = \min_{X \in \mathcal{X}_2} \max_{\|x\| = 1} \frac{\|\hat{R}x\|}{\|x\|} \leq \max_{X \in \mathcal{X}_2, \|x\| = \|\alpha u + \beta \hat{u}\| = 1} \left\| \begin{bmatrix} R & r \\ 0 & \rho \end{bmatrix} (\alpha u + \beta \hat{u}) \right\| \leq \max_{X \in \mathcal{X}_2, \|x\| = \|\alpha u + \beta \hat{u}\| = 1} \left\| \begin{bmatrix} \alpha Ry - \beta \frac{1}{N}(y^*R^{-1}r)Ry \\ 0 \end{bmatrix} \right\| + \left\| \beta \frac{1}{N} [\rho] \right\|.$$

Using the fact that $|\alpha| \leq 1$, $\|\beta\| \leq 1 \leq N$ and $|y^*R^{-1}r| \leq \|R^{-1}r\|$, we finally derive

$$\sigma_{k-1}(\hat{R}) \leq \sigma_{k-1}(R)(1 + \|R^{-1}r\|) + |\rho|. \quad \Box$$

Lemma 4.1 says that if $|\rho| \ll \sigma_{k-1}(\hat{R})$ and if $\kappa$ is not too large, then $\sigma_{k-1}(R) = O(\sigma_{k-1}(\hat{R}))$. I.e., the smallest singular value of the leading submatrix $R$ is approximately equal to the second smallest singular value of $\hat{R}$. In the following theorem, we show how the error $S_k$ can cancel the filtering property and how a possible error on the computed $Q$ matrix might influence the accuracy of the filter as well.
\textbf{Theorem 4.2.} Given an inexact RKS relation (18), with \( \| (A - \mu_k B)^{-1} S_k \| \leq \varepsilon \); given the QR decomposition \( \hat{Q} R = kK - \beta H_k \), with \( Q \) the computed approximation of \( Q = Q + \Delta Q \) such that \( \| QR - (kK - \beta H_k) \| = \| \Delta Q R \| \) is small. Then there exists a matrix \( P \) of full rank such that \( V_k^+ = (A - \mu_k B)^{-1}(\alpha A - \beta B)V_k P + E \), with \( \| E \| \leq \varepsilon \| \mu_k - \beta \| \sigma_k^{-1}(R) + \| \Delta Q \| \). Moreover, \( \| \Delta Q \| \leq \Omega \| R \| \sigma_{k-1}^{-1}(R) u \), for some \( \Omega \geq 0 \).

\textbf{Proof.} Given Eq. (18) and Lemma 2.1, we can write \( (A - \mu_k B)V_{k+1}(\alpha K - \beta H_k) = (\alpha A - \beta B)V_k(K_k - \mu_k H_k) + (\alpha \mu_k - \beta)S_k \) \( = (\alpha A - \beta B)V_k(K_k - \mu_k H_k) + (\alpha \mu_k - \beta)S_k \).

If we fill in Eq. (10) then we get, \( (A - \mu_k B)V_{k+1}\hat{Q} R = (\alpha A - \beta B)V_k(K_k - \mu_k H_k) + (\alpha \mu_k - \beta)S_k \).

Multiplying this equation on the left by \( (A - \mu_k B)^{-1} \) and on the right by \( R^{-1} \), results in \( V_{k+1}(Q + \Delta Q) = (A - \mu_k B)^{-1}(\alpha A - \beta B)V_k([K_k - \mu_k H_k]R^{-1}) + (\alpha \mu_k - \beta)(A - \mu_k B)^{-1}S_k R^{-1} \),

with \( \|((\alpha \mu_k - \beta)(A - \mu_k B)^{-1}S_k R^{-1})\| \leq \varepsilon \| \alpha \mu_k - \beta \| \sigma_k^{-1}(R) \). In [1], it is proved that \( \| \Delta Q \| \leq \sqrt{2} \| R \| \| R_{k-1}^{-1} \| \| u \| + O(u^2) \), where \( R_{k-1} \) is the \( k - 1 \times k - 1 \) leading submatrix of \( R \). If we neglect the second order term in \( u \) and if we combine this result with Lemma 4.1, then using \( \rho = (R)_{k,k} \) and \( \| R_{k-1}^{-1} \| = \sigma_{k-1}^{-1}(R_{k-1}) \), we can bound

\[ \frac{\kappa}{\sigma_{k-1}(R) - |\rho|} \geq \sigma_{k-1}^{-1}(R_{k-1}) \]

with \( \kappa \) as in Lemma 4.1. and \( |\rho|/\sigma_{k-1}(R) \approx 1 \). With the first order approximation \( \kappa/(\sigma_{k-1}(R) - |\rho|) = \kappa/\sigma_{k-1}(R)(1 + |\rho|/\sigma_{k-1}(R) + O(|\rho|^2/\sigma_{k-1}^2(R))) \approx 2\kappa/\sigma_{k-1}(R) \), we derive then

\[ \| \Delta Q \| \leq \sqrt{2} \| R \| 2\kappa \sigma_{k-1}^{-1}(R) u. \]

Thus, the error matrix \( E \) is given by the sum of both errors:

\[ E = (\alpha \mu_k - \beta)(A - \mu_k B)^{-1}S_k R^{-1} - V_{k+1} \Delta Q, \]

hence

\[ \| E \| \leq \| (\alpha \mu_k - \beta)(A - \mu_k B)^{-1}S_k R^{-1} \| + \| \Delta Q \|, \]

which proves the theorem. \( \square \)

Theorem 4.2 says that when an RKS relation is restarted and it has an \( R \) that is almost singular, then the filtering property can be lost. However, the influence of the (possibly small) error matrix \( S_k \) will dominate the inaccuracy of the computed matrix \( Q \), assuming that \( 0 \approx \sigma_k(R) \approx \sigma_{k-1}(R) \). Table 2 illustrates that this assumption is realistic to make. Notice that the results of Theorem 4.2 do not depend on the computation of \( Z \).

Let us consider a case where \( \| R^{-1} \| \) can be large. This means that \( R \) is nearly singular, so there exists a vector \( z \) of unit length, such that

\[ \| Rz \| = \| zKz - \beta Hz \| \approx 0. \]
If \( \beta/\alpha = \theta_k \), and \( z \) is equal to the corresponding eigenvector of \((K_k, H_k)\), then we get
\[
\sigma_k(R) \leq \| Rz \| = (\alpha \mu_k - \beta) |\eta_{k}z_{k,k}|.
\]
A small \( |\eta_{k}z_{k,k}| \), corresponds, following (9), to the residual of an eigenvector that has converged well. Therefore, using IFRKS to remove a converged eigenvector from the subspace \( V_{k+1} \) seems not to be a good idea, because the filtering will be inaccurate. A second filtering step can then be necessary.

Unlike the problems with the computation of \( Z \), this problem cannot be cured by switching to a different set of \( K_k \) and \( H_k \) matrices. In this case, a more explicit procedure should be employed, see Section 5. However, if condition (11) is fulfilled to computational accuracy and if \( Z \) is orthogonal, then the norm of the ‘new’ error matrix \( S_k \) will be of the same order as \( \| S_k \| \), since \( \| S_k \| \approx \| S_k Z \| = \| S_k \| \| Z \| = \| S_k \| \). So, although the filtering can fail, the error on the RKS relation will not grow. If some convergence is lost, then it can be recovered.

There is a second possibility to have a large \( \| R^{-1} \| \). If \( H_k \) has linearly dependent columns, then (in general) \( K_k \) must be singular too, since both sides of (2) must have the same rank and share the same null space. Any combination of \( K_k \) and \( H_k \) will then have a singular \( R \). Moreover, the small eigenvalue problem will have an arbitrary eigenvalue. We can expect a similar behaviour if \( H_k \) and \( K_k \) have a small singular value and \( R \) becomes near-singular. This situation must be avoided by the RKS algorithm, since it will lead to wrong results that, unfortunately, are hard to identify. Therefore, it is a good idea to normalise the columns of \( H_k \) to unit norm, in order to guard this problem: the possible near-singularity of \( H_k \) will then correspond to a very small subdiagonal element. When this type of singularity occurs, it seems best to redo the last RKS step with different parameters.

5. Deflating a converged eigenvector

There exist a straightforward alternative in order to reduce the size of Eq. (2). One can simply delete the \( p \) rightmost columns. Due to the fact that \( K_k \) and \( H_k \) are upper Hessenberg, this will result in a similar equation of a smaller dimension
\[
AV_{k-p+1}H_{k-p} = BV_{k-p+1}K_{k-p}.
\]
Clearly, no filtering is involved with this operation. Since the most recent computational work – and probably the most interesting work too – is located in the right part of Eq. (2), we must reorganise the equation before truncating it. If we are able to perform an efficient reorganisation of the subspace basis \( V_{k+1} \), then this method will be an alternative to the implicit restart. We showed in the previous section that the implicit filtering is not able to remove a converged eigenvector, so this approach will be especially valuable when such an eigenvector must be deflated. The next lemma shows some intermediate results that will be used to prove the correctness of a possible deflation procedure.

Lemma 5.1. Given an RKS triple \((V_{k+1}, H_k, K_k)\), and a scalar \( \delta \) that is no eigenvalue of \((A,B)\). Say that \( U \in \mathbb{C}^{k+1 \times k} \) is an orthogonal matrix such that \( \mathcal{R}(U) = \mathcal{R}(K_k - \delta H_k) \) and suppose that \( U^*H_k \) has full rank. Consider the generalised Schur decomposition
\[
U^*K_k\tilde{Z} = \tilde{Q}T_k \quad \text{and} \quad U^*H_k\tilde{Z} = \tilde{Q}T_H,
\]
Thus, it holds that

\[ R_{i} \]

\[ \text{Ritz values} \]

where

\[ (i) \]

\[ \text{It is easy to see that, since} \]

\[ u^{*}U = 0 \]

\[ \text{and} \]

\[ \mathcal{A}(U) = \mathcal{A}(K - \delta H) \]

\[ \text{we have} \]

\[ u^{*}(K - \delta H) = 0. \]

Thus, it holds that

\[ g_{k} = u^{*}K_{k}\tilde{Z} = u^{*}(K - \delta H)\tilde{Z} + \delta u^{*}H_{k}\tilde{Z} = \delta u^{*}H_{k}\tilde{Z} = \delta g_{H}. \]

(ii) This is a well-known result that can be seen as follows. From (19), and since \( z_{i} \) solves \( (U^{*}K - 0, U^{*}H)z_{i} = 0 \), it holds that

\[ (U^{*}K - 0, U^{*}H)\tilde{Z}\tilde{Z}^{-1}z_{i} = \tilde{Q}(T_{-} - 0, T_{H})\tilde{Z}^{-1}z_{i} = 0, \]

where \( (T_{-} - 0, T_{H})z_{i} = 0 \). Hence, the last \( k - i \) elements of \( \tilde{Z}^{-1}z_{i} \) will be zero: \( \tilde{Z}^{-1}z_{i} = I_{k}, \tilde{z}_{i} \).

(iii) Say that \( \tilde{Q} = [\tilde{Q}_{k-p} \tilde{Q}_{p}] \), with \( \tilde{Q}_{p} \in \mathbb{C}^{k+1 \times p} \), then for the space spanned by the first \( k - p \) Ritz vectors, it holds that

\[ \mathcal{A}(y_{1}, \ldots, y_{k-p}) = \mathcal{A}(V_{k+1}(I - U\tilde{Q}_{p}\tilde{Q}_{p}^{*}U^{*})H_{k}). \]

(iv) Say that \( T_{H}^{(k-p)} \) is the \( k - p \times k - p \) leading submatrix of \( T_{H} \). The Harmonic Ritz pairs of \( B^{-1}(A - \delta B) \) with respect to

\[ \mathcal{A}

\[ V_{k+1}[U\tilde{Q}_{k-p}, u]

\[ T_{H}^{(k-p)}

\[ g_{H}I_{k-k-p} \]

are equal to \( (0, y_{i}), \) \( i = 1, \ldots, k - p \), i.e. the \( k - p \) first ‘old’ Harmonic Ritz pairs with respect to \( \mathcal{A}(V_{k+1}H_{k}). \)

Proof. (i) It is easy to see that, since \( u^{*}U = 0 \) and \( \mathcal{A}(U) = \mathcal{A}(K - \delta H) \), we have \( u^{*}(K - \delta H) = 0. \)

Thus, it holds that

\[ g_{k} = u^{*}K_{k}\tilde{Z} = u^{*}(K - \delta H)\tilde{Z} + \delta u^{*}H_{k}\tilde{Z} = \delta u^{*}H_{k}\tilde{Z} = \delta g_{H}. \]

(ii) This is a well-known result that can be seen as follows. From (19), and since \( z_{i} \) solves \( (U^{*}K - 0, U^{*}H)z_{i} = 0 \), it holds that

\[ (U^{*}K - 0, U^{*}H)\tilde{Z}\tilde{Z}^{-1}z_{i} = \tilde{Q}(T_{-} - 0, T_{H})\tilde{Z}^{-1}z_{i} = 0, \]

where \( (T_{-} - 0, T_{H})z_{i} = 0 \). Hence, the last \( k - i \) elements of \( \tilde{Z}^{-1}z_{i} \) will be zero: \( \tilde{Z}^{-1}z_{i} = I_{k}, \tilde{z}_{i} \).

(iii) Say that \( T_{H}^{(p)} \) is the \( p \times p \) lower-right submatrix of \( T_{H} \). We derive from (ii) that \( y_{i} = V_{k+1}H_{i}z_{i} = V_{k+1}H_{i}\tilde{Z}I_{k}, \tilde{z}_{i} \), with \( i \leq k - p \). Then

\[ (V_{k+1}U\tilde{Q}_{p})y_{i} = \tilde{Q}_{p}^{*}U^{*}H_{i}\tilde{Z}I_{k}, \tilde{z}_{i} = [0 \ T_{H}^{(p)}]I_{k}, \tilde{z}_{i} = 0, \]

since \( i \leq k - p \). This property holds for each \( i = 1, \ldots, k - p \), so \( (V_{k+1}U\tilde{Q}_{p})[y_{1} \cdots y_{k-p}] = 0 \). The proof is completed by noticing that

\[ \mathcal{A}(V_{k+1}H_{k}) = \mathcal{A}(V_{k+1}H_{k}[z_{1} \cdots z_{k}]) = \mathcal{A}(y_{1}, \ldots, y_{k}), \text{ cfr. (7), hence} \]

\[ \mathcal{A}(y_{1}, \ldots, y_{k-p}) \subset \mathcal{A}(I - (V_{k+1}U\tilde{Q}_{p})(V_{k+1}U\tilde{Q}_{p}^{*})[y_{1} \cdots y_{k}]) \]

\[ = \mathcal{A}(V_{k+1}(I - U\tilde{Q}_{p}\tilde{Q}_{p}^{*}U^{*})H_{k}). \]

Since both subspaces have the same dimension, they must be equal.

(iv) From \( [U\tilde{Q}, u][U\tilde{Q}, u]^{*} = U\tilde{Q}_{p}(U\tilde{Q}_{p})^{*} + U\tilde{Q}_{k-p}(U\tilde{Q}_{k-p})^{*} + uu^{*} = I \), we derive

\[ V_{k+1}(I - U\tilde{Q}_{p}\tilde{Q}_{p}^{*}U^{*})H_{k} \]

\[ = V_{k+1}(U\tilde{Q}_{k-p}\tilde{Q}_{k-p}^{*}U^{*} + uu^{*})H_{k}\tilde{Z} = V_{k+1}(U\tilde{Q}_{k-p}[T_{H}^{(k-p)} \ast] + uu^{*}H_{k}). \]
Since $T_H^{(k-p)}$ has nonzero diagonal elements, it has full rank. Hence, the range of this matrix is spanned by its $k - p$ first columns, which are linearly independent:

$$\mathcal{R}(V_{k+1}(I - U \tilde{Q}_p \tilde{Q}_p^* U^*) H_k) = \mathcal{R}(V_{k+1}(U \tilde{Q}_{k-p} T_H^{(k-p)} + u g_H I_{k,k-p})).$$

Following (iii), this subspace is spanned by $[y_1 \cdots y_{k-p}]$, which are the mentioned Harmonic Ritz vectors.

The generalised Schur decomposition can be reordered such that the first $k - p$ Ritz values emerge in some chosen sequence. This technique was already used in [3]. If there are multiple Ritz values, then one (or more) of the corresponding vectors is removed, depending on the ordering of Eq. (19).

Using Lemma 5.1, we can now derive how the RKS relation can be transformed in a smaller RKS relation without losing the first $k - p$ Harmonic Ritz vectors. Before proving this theorem, we recall that if we choose $\delta \equiv \mu_k$, then $[U \ u] = I_{k+1}$, which simplifies the notation a lot.

We assumed that $\mathcal{R}(V_{k+1})$ does not contain an exact eigenvector of $(A,B)$. Therefore, we may assume that the first entry of $g_H$ is nonzero. It can be seen from Eq. (21) that if $g_H e_1 = 0$, then $y \equiv V_{k+1} U \tilde{Q} e_1$ would be an eigenvector of $(A,B)$, corresponding to the eigenvalue $\lambda = \theta_1$, so we may conclude that $g_H e_1 \neq 0$.

**Theorem 5.2.** Under the conditions of Lemma 5.1, set $Q \equiv [U \tilde{Q}_{k-p}, u]$, then there exists an orthogonal upper Hessenberg matrix $F \in \mathbb{C}^{k-p \times k-p}$, such that with $Z \equiv \tilde{Z} I_{k-p} F$ and

$$V_{k-p+1}^+ \equiv V_{k+1} Q, \quad H_{k-p}^+ \equiv Q^* H_k Z, \quad K_{k-p}^+ \equiv Q^* K_k Z,$$

$(V_{k-p+1}^+, H_{k-p}^+, K_{k-p}^+)$ is an RKS triple of order $k - p$ for $(A,B)$. Moreover, the Harmonic Ritz pairs of this triple with respect to $B^{-1}(A - \delta B)$ are given by $(\tilde{\theta}_i^+, \tilde{y}_i^+) = (\theta_i, y_i), i = 1, \ldots, k - p$.

**Proof.** We verify the three conditions of Definition 2.1. The matrix $V_{k-p+1}^+$ is clearly orthogonal. Following Lemma 3.4, there exists an orthogonal upper Hessenberg $F \in \mathbb{C}^{k-p \times k-p}$ such that $g_H I_{k,k-p} F = [0 \cdots 0 \ 1]$. Since $g_H e_1 \neq 0$, $F$ is unreduced. By construction, $H_{k-p}^+$ and $K_{k-p}^+$ are upper Hessenberg, since

$$Q^* H_k Z = \begin{bmatrix} T_H^{(k-p)} \\
 g_H I_{k,k-p} \end{bmatrix} \ F \quad \text{and} \quad Q^* K_k Z = \begin{bmatrix} T_k^{(k-p)} \\
 \delta g_H I_{k,k-p} \end{bmatrix} \ F,$$

(20)

with $F$ upper Hessenberg.

(i) We have to prove that $AV_{k+1} H_k Z = BV_{k+1} K_k Z$. If we insert $[U \tilde{Q}, u][U \tilde{Q}, u]^*$ in Eq. (2) and multiply it on the right by $\tilde{Z} I_{k,k-p}$, then

$$AV_{k+1} [U \tilde{Q}, u][U \tilde{Q}, u]^* H_k \tilde{Z} I_{k,k-p} = BV_{k+1} [U \tilde{Q}, u][U \tilde{Q}, u]^* K_k \tilde{Z} I_{k,k-p}.$$ 

If we apply Eq. (19), then this corresponds to

$$AV_{k+1} [U \tilde{Q}, u] \begin{bmatrix} T_H^{(k-p)} \\
 0 \\
 g_H I_{k,k-p} \end{bmatrix} = BV_{k+1} [U \tilde{Q}, u] \begin{bmatrix} T_k^{(k-p)} \\
 0 \\
 g_k I_{k,k-p} \end{bmatrix}$$

(21)
\[ \Rightarrow AV_{k+1}[U\hat{Q}_{k-p}, u] \begin{bmatrix} \mathcal{T}_{H}^{(k-p)} \\ \delta g_{Hk,k-p} \end{bmatrix} = BV_{k+1}[U\hat{Q}_{k-p}, u] \begin{bmatrix} \mathcal{T}_{H}^{(k-p)} \\ \delta g_{Hk,k-p} \end{bmatrix}. \] (22)

If we multiply this equation on the right by \( F \), then we find by (20) that (2) is fulfilled.

(ii) The matrix \( H_{k-p}^{+} \) is unreduced, since none of the \( (T_{H})_{i,j} \) is zero and \( F \) is unreduced. From \( (20) \), it can be derived that \( \mu_{i}^{+} = \theta_{i+1} \) for \( i = 1, \ldots, k - p - 1 \) and \( \mu_{k-p}^{+} = \delta \).

(iii) Since none of the \( \theta_{i} \) is an eigenvalue of \((A, B)\) and neither is \( \delta \), it follows that \( \mu_{i}^{+} \) is no eigenvalue of \((A, B)\), for \( i = 1, \ldots, k - p \).

Finally, it follows from Lemma 5.1(iv) that the ‘old’ and the ‘new’ Harmonic Ritz pairs are equal: \((\theta_{i}, y_{i}^{+}) = (\theta_{i}, y_{i}), i = 1, \ldots, k - p \). \( \Box \)

Theorem 5.2 shows that in order to reduce the dimension of the RKS relation, without changing the approximate eigenvalues \( \theta_{i} \), we must restrict \( V_{k+1}U\hat{Q} \) to its first \( k - p \) columns and add the vector \( V_{k+1}u \). Algorithm 3 summarises these results.

**Algorithm 3.** Reordering and truncating the RKS relation

**Input:** The RKS triple \((V_{k+1}, H_{k}, K_{k})\), a parameter \( \delta \) and an integer \( p < k \).

**Output:** The RKS triple \((V_{k+1}^{\prime}, H_{k-p}^{+}, K_{k-p}^{+})\).

1. Compute the unitary matrix \([U u] \) such that \( \mathcal{R}(U) = \mathcal{R}(K_{k} - \delta H_{k}) \).
2. Compute the generalised Schur decomposition
   \[ U^{*}K_{k}Z = \hat{O}T_{k} \quad \text{and} \quad U^{*}H_{k}Z = \hat{O}T_{H}, \] (23)
   such that \( \theta_{i} = (T_{H})_{i,j}/(T_{H})_{j,j} \), \( i = 1, \ldots, k - p \) are ‘wanted’ Ritz values.
3. Compute \( \hat{F} \in C^{k-p \times k-p+1} \) such that if \( g_{H} \equiv u^{*}H_{k}Z \), then \( g_{H}H_{k,k-p}F = 0 \).
   Set \( F = [\hat{F}, (g_{H}H_{k,k-p})^{*}] \).
4. Set \( V_{k-p+1}^{+} = V_{k+1}[U\hat{O}I_{k,k-p}, u], \)
   \( H_{k-p}^{+} = [U\hat{O}I_{k,k-p}, u]^{*}H_{k}ZI_{k,k-p}F \) and
   \( K_{k-p}^{+} = [U\hat{O}I_{k,k-p}, u]^{*}K_{k}ZI_{k,k-p}F \).

If some eigenvectors have converged well, then the reordered equation will be *nearly reduced*. We could assume that the first vectors in \( V_{k+1}^{+} \) contain an invariant subspace of \((A, B)\), without making a large error. The following theorem illustrates this, assuming that \( l \) eigenvectors have converged.

**Theorem 5.3.** Suppose that Theorem 5.2 is applied with \( p=0 \) and suppose that in Eq. (19), \((T_{H})_{i,j}/(T_{H})_{j,j} = \theta_{i} \) are ordered such that \( \theta_{1}, \ldots, \theta_{l} \) are well converged eigenvalues. If

\[ \| (AV_{k+1}QT_{H} - BV_{k+1}QT_{H})I_{k,l} \| < \varepsilon, \] (24)

with \( \varepsilon \) small and

\[ \| (AV_{k+1}QT_{H} - BV_{k+1}QT_{H})I_{k,l+1} \| > \tau \gg \varepsilon, \] (25)

then the first \( l \) columns of \( H_{k}^{+} \) are near to an upper triangular matrix, i.e.

\[ H_{k}^{+}I_{k,l} = \begin{bmatrix} T_{H} \\ 0 \end{bmatrix} + E, \quad \text{with} \quad \| E \| \leq \varepsilon + O \left( \frac{\varepsilon^{2}}{\tau^{2}} \right). \] (26)
Then we show that for removing spurious eigenvalues or for acceleration, then we must use IFRKS.

Say that \( g_H = [g^{(i)} \gamma g^{(k-l-1)}] \) is equal to \( u^*H_k \tilde{Z} \), with \( g^{(i)} \in \mathbb{C}^{1 \times l} \). Without loss of generality, we suppose that \( g_H e_1 \) is nonzero. Otherwise, the first columns of \( Z \) will be exactly equal to those of \( \tilde{Z} \) and the corresponding columns of \( E_Z \) are zero.

We divide the proof in two parts. First, we show that \( \| g^{(i)} \| / |\gamma| \ll \varepsilon / (\tau - \varepsilon) = (\varepsilon / \tau) + O(\varepsilon^2 / \tau^2) \).

(A) If we combine \( QQ^* = I \), with Eq. (2), multiplied on the right by \( \tilde{Z} \), then we find that

\[
A(V_{k+1}Q) \begin{bmatrix} T_0 \\ g_H \end{bmatrix} = B(V_{k+1}Q) \begin{bmatrix} T_k \\ g_k \end{bmatrix},
\]

and hence

\[
(A(V_{k+1}U \tilde{Q})T_0 - B(V_{k+1}U \tilde{Q})T_k) = -(A - \delta B)V_{k+1}u g_H.
\]

If we take the first \( l \) columns of Eq. (28) and fill them in Eq. (24), then

\[
\|(A - \delta B)V_{k+1}u g^{(i)}\| = \|(A - \delta B)V_{k+1}u\| \|g^{(i)}\| = \|A - \delta B\| \|g^{(i)}\| < \varepsilon
\]

(For the outer product of two vectors \( a, b \), it holds that \( \|ab^*\| = \|a\| \|b\| \)). Similarly, combining the first \( l + 1 \) columns of Eq. (28) with Eq. (25) results in

\[
\|A - \delta B\| \|g^{(i)}\| + |\gamma| \|A - \delta B\| \|[g^{(i)} \gamma]\| = \|(A - \delta B)V_{k+1}u g^{(i)}\gamma\| > \tau.
\]

By dividing these equations, we find that

\[
1 + \frac{|\gamma|}{\|g^{(i)}\|} > \frac{\tau}{\varepsilon}, \text{ hence, } \frac{\|g^{(i)}\|}{|\gamma|} < \frac{\varepsilon}{\tau - \varepsilon} = \frac{\varepsilon}{\tau} + O \left( \frac{\varepsilon^2}{\tau^2} \right).
\]

(B) Define \( \tilde{F} \in \mathbb{C}^{l+1 \times l} \) as

\[
\tilde{F} \equiv \begin{bmatrix} I \\ \gamma I \\ -g^{(i)} \gamma \end{bmatrix},
\]

then \( g^{(i)} \tilde{F} = 0 \) and the columns of \( \tilde{F} \) have almost unit norm. As a result, with \( E_Z \equiv \tilde{F} - I_{l \times l} \),

\[
\|E_Z\| = \left\| \begin{bmatrix} I \\ \gamma I \\ -g^{(i)} \gamma \end{bmatrix} - \begin{bmatrix} I \\ 0 \end{bmatrix} \right\| = \|g^{(i)}\|/|\gamma|.
\]

Therefore, Eq. (27) holds, with \( \|E_Z\| = \|g^{(i)}\|/|\gamma| \ll (\varepsilon / \tau) + O(\varepsilon^2 / \tau^2) \). This completes the proof.

If the first \( l \) subdiagonal elements of \( H_k \) and \( K_k \) are of the order of the machine precision, then we could set these elements explicitly to zero without introducing a relevant error. We work further with the unreduced upper Hessenberg lower-right part of \( H_k \) and \( K_k \), but in step 1.3 of Algorithm 1, each new iteration vector is also orthogonalised to the converged Schur vectors.

Truncation of the relation can be used to restart RKS by removing more than one Ritz vector in one step. It is a valuable alternative to IFRKS. On the other hand, only Ritz vectors can be removed with this method and no implicit filtering as in (1) can be done. If the restart is used for validation, for removing spurious eigenvalues or for acceleration, then we must use IFRKS.
6. Example

In this example, we show how the results that are presented in this text can be used in a non-academic context. Let us recall the example in [2].

The example comes from a model of viscous free-surface fluid flow on a tilted plane [6]. The Navier–Stokes equations were discretised by a finite element approach leading to an eigenvalue problem $Ax = \lambda Bx$ of size $n = 536$. The matrices $A$ and $B$ are nonsymmetric, $B$ is singular, ($B$ has rank 429) and $A$ is not. The goal is to find the rightmost eigenvalues used for the stability analysis of a steady state solution of the Navier–Stokes equations. These eigenvalues are

$$
\theta_1 = -9.48831,
$$

$$
\theta_{2,3} = -11.6062 \pm 14.6602i,
$$

$$
\theta_{4,5} = -15.9689 \pm 3.23426i,
$$

$$
\theta_{6,7} = -21.8621 \pm 36.3798i.
$$

The singularity of the matrix $B$ causes $Ax = \lambda Bx$ to have an infinite eigenvalue, which is in this case also defective. When we apply an iterative eigenvalue solver to this problem, then this infinite eigenvalue will be approximated by large, spurious eigenvalues. These eigenvalues can cause numerical problems (if they are very large) or they can mislead the algorithm (if they can not be identified as being ‘spurious’).

We avoid the calculation of spurious eigenvalues by including in the algorithm repeatedly a filtering step. We use the property that the eigenvectors corresponding to the infinite eigenvalue, lie in the nullspace of $B$ and $(A - \mu B)^{-1}Bv$. We remove them by applying IFRKS with $(\alpha, \beta) \equiv (0, 1)$ such that

$$
R(V_k^+) = R((A - \mu_k B)^{-1}BV_k).
$$

Algorithm 4 shows the approach that is used in this example. The algorithm performs first $k_{\text{start}} = 10$ iterations of RKS with a fixed initial pole $\mu_1 = -5$ in order to ‘locate’ the rightmost eigenvalues. Then the new pole $\mu$ is set to the approximate rightmost eigenvalue $\theta$, and RKS is used to approximate accurately the eigenvector. If the corresponding residual is smaller than a given tolerance $\tau_r = 1e-15$, then the eigenvector is deflated from the subspace and a new pole is set. There are some safety measurements included. As said, we filter out the spurious eigenvalues with $k_{\text{fil}} = 2$ filter steps in each loop (IFRKS is implemented with the orthogonal choice of $Z^\perp$). If the dimension of the RKS triple $(V,H,K)$ exceeds a maximal number $k_{\text{max}} = 20$, then the size is brought back to $k_{\text{start}}$, using the IFRKS algorithm with exact shifts removing the leftmost Ritz values. If the residual that corresponds to an approximate eigenvalue is smaller than the tolerance $\tau_r = 1e - 5$, then this eigenvalue is not selected as the new pole, or it is perturbed with an error of order $O(\tau_r)$ – see step 2.2.4 of the algorithm. This rule was included in order to avoid the occurrence of very ill-conditioned systems in the RKS steps. Experiments showed that if, by ‘accident’, a pole was chosen very close to an eigenvalue, then a considerable error on the RKS relation was introduced. This error depends of course on the method that is used for solving the linear system.
Table 3
The table shows for each detected eigenvalue the number of additional iteration steps that were needed to approximate the eigenvalues accurately. The number of iteration steps (outer loops) are shown for the three choices of Z. The first column shows the residual of the eigenvalue after detection. The third column shows the error on the RKS relation after the computation of the eigenvectors. The last columns show the final accuracy of the eigenvalue and the final residual.

| θ_i | Init. residual norm | No. of iterations | AVH − BVK || H || | λ_i − θ_i | |Ay_i − θ_iBy_i|| |
| --- | --- | --- | --- | --- | --- | --- | --- |
| −9.48831 | 1.4e − 10 | 3 (4, 4) | 7.1e − 14 | 6.5e − 8 | 1.4e − 13 |
| −11.6062 − 14.6602i | 4.8e − 7 | 3 (3, 2) | 8.8e − 14 | 1.3e − 5 | 1.1e − 15 |
| −11.6062 + 14.6602i | 1.0e − 7 | 3 (2, 1) | 6.5e − 14 | 2.0e − 6 | 2.5e − 15 |
| −15.9689 + 3.23426i | 1.8e − 10 | 2 (2, !) | 6.0e − 14 | 1.7e − 7 | 1.3e − 14 |
| −15.9689 − 3.23426i | 2.1e − 10 | 2 (2, !) | 4.5e − 14 | 2.4e − 7 | 9.4e − 14 |

The results of the experiment are displayed in Table 3. They show that the algorithm is able to find several eigenvalues of a generalised eigenvalue problem. The eigenvalues that were identified in the first set of iterations can be efficiently approximated: it took about 2 or 3 extra iteration steps for each eigenvalue. The combined approach of deflating the converged eigenvalues with IFRKS – for filtering and restarting when there is no convergence – seems to be efficient and robust. Two possible difficulties were detected. First, a good estimator for the residual norm of an approximate eigenpair is needed. An algorithm that finds more than one eigenvalue must contain a reliable strategy to decide whether an eigenvalue has converged or not. The algorithm must also decide which approximate eigenvalue is a good pole. It is possible that a spurious eigenvalue is found that looks like a converged, true eigenvalue. There is also no guarantee that no eigenvalue is missed. In both cases, some validation of the results is needed.

Algorithm 4. Tilted Plane

**Input:**

- μ_1 (the initial pole), k_{start} (number of initial iteration steps),
- v_1 = [1 · · · 1]^T (starting vector), k_max (maximal size of V_k),
- k_{fil} (number of filtering steps), τ_c (convergence tolerance), τ_μ (pole tolerance),
- m (number of wanted eigenvalues).

**Output:** (θ_1, y_1), . . . , (θ_m, y_m)

1. Perform k_{start} steps of RKS with pole μ_1.
   
   Set i = 1.

2. While i ≤ m, do
   2.1. Compute (θ_i, y_i) and the residual ||r(y_i)||.
   2.2. If ||r(y_i)|| < τ_c then
      2.2.1. Deflate the converged eigenpair and reduce the size of (V, H, K) to k_{start} + k_{fil} (using Algorithm 3).
      2.2.2. Perform k_{fil} steps of IFRKS with (α, β) = (0, 1).
2.2.3. Set $i \leftarrow i + 1$ and find the next approximate eigenvalue $\theta_i$.

2.2.4. Set $\mu \leftarrow \theta_i(1 + \tau_\mu)$.

2.3. Else if $\| r(y_i) \| < \tau_\mu$ then
   2.3.1. Perform $k_\text{fil}$ steps of IFRKS with $(\alpha, \beta) = (0, 1)$.
   2.3.2. Reduce the size of $(V, H, K)$ to $k_{\text{start}}$ by removing the leftmost Ritz values using IFRKS with exact shifts.

2.4. Else
   2.4.1. Set $\mu \leftarrow \theta_i$.
   2.4.2. Perform $k_\text{fil}$ steps of IFRKS with $(\alpha, \beta) = (0, 1)$.
   2.4.3. Reduce the size of $(V, H, K)$ to $k_{\text{start}}$ by removing the leftmost Ritz values using IFRKS with exact shifts.
   End If.

2.5. Perform RKS with shift $\mu$ until $\| r(y_i) \| < \tau_v$ or the size of $(V, H, K)$ is equal to $k_{\text{max}}$.

We compared the results with two different values for $k_\text{fil}$. First, we ran the same algorithm without the intermediate filtering step. This algorithm didn’t return any useful results: it found some eigenvalue of order $1e + 6$. Then we set the parameter $k_\text{fil}$ to 1, instead of $k_\text{fil} = 2$, which gave the same results as in Table 3. Obviously, our choice of $k_\text{fil} = 2$ was too cautious. In a second experiment, we replaced the deflation algorithm (Algorithm 3) by IFRKS with exact shifts. The algorithm then returned correct results, but it needed much more iteration steps (about 10 times more), since at each deflation step, much of the accuracy was lost.

Finally, we tried the different computations of $Z$. The results were very comparable for $Z^\perp$ and $Z^\perp$. However, if we used $Z^{T^{-1}}$, then the algorithm could not find more that three correct eigenvalues. After the third eigenvalue, the accumulated error caused the algorithm to pick a wrong shift and a bad, spurious approximation for the first eigenvalue was found.

7. Conclusion

In this text, we showed how the RKS relation can be restarted implicitly in different ways. The first option was to compute the matrix $Z$ while applying the inverse of the $T_k$ matrix (implicitly). Another option was to compute $Z$ directly such that it is orthogonal. Both options lead to comparable results, however the orthogonal $Z^\perp$ is preferred, since it preserves the accuracy of the RKS relation better in cases where an eigenvector has converged well.

 Restarting the RKS relation can be combined with an implicit filtering of the subspace that is spanned by $V_k$. When the RKS relation contains an eigenvector that has almost converged, then the matrix $QR = \alpha K_k - \beta H_k$ can be nearly singular. This causes the failure of the filtering step and, by possible forward instabilities in the QR factorisation, brings about an inaccurate restart. We showed that the influence of the inaccurate computation of $Q$ on the filtering error is smaller than the influence of the singularity. However, even a ‘less inaccurate’ $Q$ can cause already loss of convergence.
As a possible solution to this problem, we showed how the RKS relation can be transformed and truncated. This approach can be used to deflate converged eigenvectors, whether they are wanted or unwanted. Therefore, this algorithm can be used as a restarting procedure for RKS, but it does not come along with an implicit filter. Both approaches provide us with the building blocks of a robust, iterative RKS implementation that solves partially a generalised eigenvalue problem.

Acknowledgements

This research was supported by the National Fund for Scientific Research (NFWO), project Lanczos, grant #2.0042.93 and by the Human Capital and Mobility project ROLLS of the European Community under contract ERBCHRXCT930416. Lots of thanks goes at to the patient referees.

References


