Electronic Transactions on Numerical Analysis.
Volume 43, pp. 100-124, 2014.

ISSN 1068-9613.

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

COMPUTING APPROXIMATE (BLOCK) RATIONAL KRYLOV SUBSPACES WITHOUT EXPLICIT INVERSION WITH EXTENSIONS TO SYMMETRIC MATRICES*

THOMAS MACH†, MIROSLAV S. PRANIĆ‡, AND RAF VANDEBRIL†

Abstract. It has been shown that approximate extended Krylov subspaces can be computed, under certain assumptions, without any explicit inversion or system solves. Instead, the vectors spanning the extended Krylov space are retrieved in an implicit way, via unitary similarity transformations, from an enlarged Krylov subspace. In this paper this approach is generalized to rational Krylov subspaces, which aside from poles at infinity and zero, also contain finite non-zero poles. Furthermore, the algorithms are generalized to deal with block rational Krylov subspaces and techniques to exploit the symmetry when working with Hermitian matrices are also presented. For each variant of the algorithm numerical experiments illustrate the power of the new approach. The experiments involve matrix functions, Ritz-value computations, and the solutions of matrix equations.

Key words. Krylov, extended Krylov, rational Krylov, iterative methods, rotations, similarity transformations

AMS subject classifications. 65F60, 65F10, 47J25, 15A16

1. Introduction.

In [17] we presented a method for computing approximate extended Krylov subspaces generated by a matrix A and vector v. This approach generates the vectors $A^{-k}v$, spanning the Krylov subspace, in an implicit way without any explicit inversion: A^{-1} or system solve: $A^{-1}v$. We showed that for several applications the approximation provides satisfying results. Here we generalize this algorithm to rational (block) Krylov subspaces, and we will show how to use and preserve symmetry when dealing with symmetric or Hermitian matrices.

Let $A \in \mathbb{C}^{n\times n}$ and $v \in \mathbb{C}^n$. The subspace

$$\mathcal{K}_m(A, v) = \operatorname{span}\left\{v, Av, A^2v, \ldots, A^{m-1}v\right\} \tag{1.1}$$

is called a *Krylov subspace*. Krylov subspaces are frequently used in various applications, typically having large datasets to be analyzed, e.g., for solving symmetric sparse indefinite systems [20], large unsymmetric systems [25], or Lyapunov equations [11]. Rational Krylov subspaces were introduced by Ruhe in [21], investigated later in [22, 23, 24], and they have been used to solve matrix equations, for instance, in the context of model order reduction; see, e.g., [1, 3, 5, 7, 9] or more recently for bilinear control systems [2]. Let $\sigma = [\sigma_1, \sigma_2, \ldots, \sigma_{m-1}]$, with $\sigma_i \in (\mathbb{C} \cup \{\infty\}) \setminus \Lambda(A)$, where $\Lambda(A)$ is the set of eigenvalues

*Received September 30, 2013. Accepted June 19, 2014. Published online on October 16, 2014. Recommended by L. Reichel. The research was partially supported by the Research Council KU Leuven, projects CREA-13-012 Can Unconventional Eigenvalue Algorithms Supersede the State of the Art, OT/11/055 Spectral Properties of Perturbed Normal Matrices and their Applications, CoE EF/05/006 Optimization in Engineering (OPTEC), and fellowship F+/13/020 Exploiting Unconventional QR-Algorithms for Fast and Accurate Computations of Roots of Polynomials, by the DFG research stipend MA 5852/1-1, by the Fund for Scientific Research–Flanders (Belgium) project G034212N Reestablishing Smoothness for Matrix Manifold Optimization via Resolution of Singularities, by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office, Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), and by the Serbian Ministry of Education and Science project #174002 Methods of Numerical and Nonlinear Analysis with Applications.

†Department of Computer Science, KU Leuven, Celestijnenlaan 200A, 3001 Leuven (Heverlee), Belgium ({thomas.mach,raf.vandebril}@cs.kuleuven.be).

‡Department Mathematics and Informatics, University of Banja Luka, M. Stojanovića, 51000 Banja Luka, Bosnia and Herzegovina (pranic77m@yahoo.com).

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

of A. Then

$$\mathcal{K}_m^{\text{rat}}(A, v, \sigma) = q_{m-1}(A)^{-1}\mathcal{K}_m(A, v), \text{ with } \quad q_{m-1}(z) = \prod_{\substack{j=1 \\ \sigma_j \neq \infty}}^{m-1} (z - \sigma_j)$$

is called a *rational Krylov subspace*. If we set all finite shifts of an $m_\ell + m_r - 1$ dimensional rational Krylov subspace to 0, then the subspace becomes

$$\mathcal{K}_{m_\ell, m_r}(A, v) = \operatorname{span}\left\{A^{-m_r+1}v, \ldots, A^{-1}v, v, Av, A^2v, \ldots, A^{m_\ell-1}v\right\},$$

which is called an *extended Krylov subspace*. Extended Krylov subspaces were investigated first by Druskin and Knizhnerman in [4]. The advantage over rational Krylov subspaces is that only one inverse, factorization, or preconditioner of A (to approximately compute $A^{-1}v$) is necessary; see, e.g., [12, 13, 15]. On the other hand the additional flexibility of different shifts in the rational Krylov case might be used to achieve the same accuracy with smaller subspaces, but for this, one needs good shifts, which recently was investigated in [10] by Güttel.

For every Krylov subspace $\mathcal{K}_m(A, v)$ of dimension m a matrix $V \in \mathbb{C}^{n\times m}$ with orthogonal columns exists, so that

$$\operatorname{span}\{V_{:,1:k}\} = \operatorname{span}\left\{v, Av, A^2v, \ldots, A^{k-1}v\right\} \quad \forall k \leq m, \tag{1.2}$$

where $V_{:,1:k}$ is MATLAB-like notation referring to the first k columns of V. It is well known that the *projected counterpart* $H := V^*AV$ of A, with V^* being the conjugate transpose of V, is of Hessenberg form, i.e., all the entries $H_{i,j}$ with $i > j + 1$ are zero [8]. Let V now be defined analogously for a rational Krylov subspace with only finite poles, $\mathcal{K}_m^{\text{rat}}(A, v, \sigma)$. In [6], Fasino showed that for A Hermitian that $H = V^*AV$ is of Hermitian diagonal-plus-semiseparable form, meaning that the submatrices $H_{1:k,k+1:n}$, for $k = 1, \ldots, n-1$, are of rank at most 1. However, if V spans an extended Krylov subspace of the form

$$\operatorname{span}\left\{v, Av, A^{-1}v, A^{-2}v, A^{-3}v, A^2v, A^3v, \ldots\right\},$$

then $H = V^*AV$ is a matrix having diagonal blocks of Hessenberg or of inverse Hessenberg form [28] (these blocks overlap), where a matrix is of inverse Hessenberg form[1] if the rank of $H_{1:k,k:n}$ is at most 1 for $k = 1, \ldots, n-1$; at the end of Section 2.1 a more precise definition of extended Hessenberg matrices is presented. In Section 2 we will describe the structure of H for rational Krylov subspaces with mixed finite and infinite poles.

The main idea of computing approximate, rational Krylov subspaces without inversion is to start with a large Krylov subspace and then apply special similarity transformations to H to bring the matrix into the extended Hessenberg plus diagonal form, the form one would get if one applied a rational Krylov algorithm directly. To achieve this form no inversions or systems solves with A or $A - \sigma_i I$ are required. At the end we keep only a small upper left part of H containing the main information. We will show that under certain assumptions the computed $\hat{H}$ and $\hat{V}$ approximate the matrices H and V obtained directly from the rational Krylov subspace, as we have already shown for extended Krylov subspaces in [17].

Block Krylov subspace methods are an extension of Krylov subspace methods, used, for instance, to solve matrix equations with right-hand sides of rank larger than one; see [11, 14].

[1] These matrices are said to be of inverse Hessenberg form, as their inverses, for nonsingular matrices, are Hessenberg matrices.

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

Instead of using only a single vector v, one uses a set of orthogonal vectors $\mathcal{V} = [v_1, v_2, \ldots, v_b]$. The block Krylov subspace then becomes

$$\begin{aligned}\mathcal{K}_m(A, \mathcal{V}) &= \operatorname{span}\left\{\mathcal{V}, A\mathcal{V}, A^2\mathcal{V}, A^3\mathcal{V}, \ldots, A^{m-1}\mathcal{V}\right\} \\ &= \operatorname{span}\left\{v_1, \ldots, v_b, Av_1, \ldots, Av_b, \ldots\right\}.\end{aligned}$$

Block Krylov subspaces can often be chosen of smaller dimension than the sum of the dimension of the Krylov subspaces $\mathcal{K}(A, v_1), \ldots, \mathcal{K}(A, v_b)$, since one uses information from $\mathcal{K}(A, v_i)$ for, e.g., the approximation of a matrix function times a vector: $f(A)v_j$. Block extended and block rational Krylov subspaces can be formed by adding negative powers of A such as $A^{-k}\mathcal{V}$ or $\prod_{j=k,\sigma_j\neq\infty}^{1}(A - \sigma_j I)^{-1}\,\mathcal{V}$. We will describe the approximation of block rational Krylov subspaces in Section 3.

If the matrix A is symmetric or Hermitian[2], then the matrix $H = V^*AV$ inherits this structure; thus H becomes tridiagonal. Exploiting the symmetry reduces the computational costs of the algorithm and is discussed in Section 4.

First we introduce the notation and review the essentials about rotators.

1.1. Preliminaries. Throughout the paper the following notation is used. We use capital letters for matrices and lower case letters for (column) vectors and indices. For scalars we use lower case Greek letters. Arbitrary entries or blocks of matrices are marked by $\times$ or by $\otimes$. Let $I_m \in \mathbb{C}^{m\times m}$ denote the identity matrix and $e_i \in \mathbb{C}^m$ stands for the ith column of I_m. We further use the following calligraphic letters: $\mathcal{O}$ for the big O notation, $\mathcal{K}$ for Krylov subspaces, $\mathcal{V}$ for subspaces, and $\mathcal{E}_k$ for the subspace spanned by the first k columns of the identity matrix.

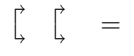
The presented algorithms rely on clever manipulations of rotators. Therefore we briefly review them. *Rotators* are equal to the identity except for a 2×2 unitary block on the diagonal of the form

$$\begin{bmatrix} \alpha & \beta \\ -\overline{\beta} & \overline{\alpha} \end{bmatrix},$$

with $|\alpha|^2 + |\beta|^2 = 1$. They are also known as *Givens* or *Jacobi rotations* [8]. To simplify the notation and be able to depict the algorithms graphically, we use $\vec{\curvearrowright}$ to depict a single rotator. The tiny arrows point to the two rows where the 2×2 block is positioned. If the rotator is applied to a matrix on the right, then the arrows also point to the two rows of the matrix that are changed. If we have a series of rotators, e.g.,

then we call the ordering of the rotators a *shape* or a *pattern* [19].

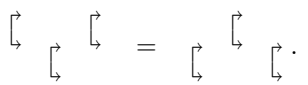
To handle rotators efficiently we need three operations: merging, turnover, and transfer of rotators through upper triangular matrices. Two rotators acting on the same rows can be *merged*, resulting in a single rotator



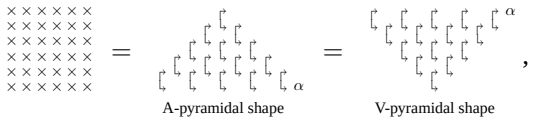
[2] In the remainder of this paper A symmetric means that A equals its conjugate transpose: $A = A^T$ for $A \in \mathbb{R}^{n\times n}$ and $A = A^*$ for $A \in \mathbb{C}^{n\times n}$.

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

Three rotations in a V-shaped sequence can be replaced by three rotations in an A-shaped sequence (and vice versa),

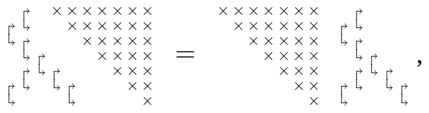


This is called a *turnover*. More generally it is possible to factor an arbitrary unitary matrix $Q \in \mathbb{C}^{n\times n}$ into $\frac{1}{2}n(n-1)$ rotators times a diagonal matrix D_α. This diagonal matrix D_α equals the identity except for a single diagonal element $\alpha = \det Q$. There are various possible patterns for arranging these rotators and the position of α in the diagonal of D_α. The A- and V-pyramidal shape, graphically visualized as

$Q =$ 

where in the schemes the diagonal matrix D_α is not shown, only the value α is depicted, having the row in which it is positioned corresponding to the diagonal position of α in D_α. The main focus is on the ordering of the rotators, the diagonal matrix D_α does not complicate matters significantly and is therefore omitted. If the pyramidal shape points up we call it an A-pyramidal shape, otherwise a V-pyramidal shape. A sequence of rotators in A-pyramidal shape can always be replaced by a sequence of rotators in V-pyramidal shape [27, Chapter 9].

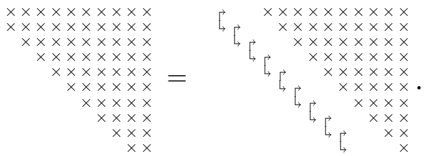
Further, one can *transfer rotators through* an upper triangular matrix. Therefore one has to apply the rotator to the upper triangular matrix, assume it is acting on rows i and $i+1$, creating thereby an unwanted non-zero entry in position $(i+1, i)$ of the upper triangular matrix. This non-zero entry can be eliminated by applying a rotator from the right, acting on columns i and $i+1$. Transferring rotators one by one, one can pass a whole pattern of rotators through an upper triangular matrix, e.g.,



thereby preserving the pattern of rotations.

In this article we will use the QR decomposition extensively. Moreover, we will factor the unitary Q as a product of rotations. If a matrix exhibits some structure, often also the pattern of rotations in Q's factorization is of a particular shape.

A *Hessenberg matrix* H is said to be *unreduced* if none of the subdiagonal entries (the elements $H_{i+1,i}$) equal zero. To shift this notion to extended Hessenberg matrices we examine their QR decompositions. The QR decomposition of a Hessenberg matrix is structured, since the unitary matrix Q is the product of $n-1$ rotators in a descending order, e.g.,



The matrix H being unreduced corresponds thus to that all rotators are different from a diagonal matrix. An *extended Hessenberg matrix* [26] is defined by its QR decomposition consisting of $n-1$ rotators acting on different rows as well, but reordered in an arbitrary, not necessarily descending, pattern; see, e.g., the left term in the right-hand side of (2.5). In

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

correspondence with the Hessenberg case we call an extended Hessenberg matrix *unreduced* if all rotators are non-diagonal.

2. Rational Krylov subspaces. In [17] we have shown how to compute an approximate extended Krylov subspace. We generalize this, starting with the simplest case: the rational Krylov subspace for an arbitrary unstructured matrix. We further discuss block Krylov subspaces and the special adaptions to symmetric matrices. The main difference to the algorithm for extended Krylov subspaces is that finite non-zero poles are present and need to be introduced. This affects the structure of the projected counterpart $H = V^*AV$ and the algorithm. Further, we need an adaption of the implicit-Q-theorem [17, Theorem 3.5]; see Theorem 2.1.

2.1. Structure of the projected counterpart in the rational Krylov setting. Let $\sigma = [\sigma_1, \sigma_2, \ldots, \sigma_{m-1}]$, with $\sigma_i \in (\mathbb{C} \cup \{\infty\}) \setminus \Lambda(A)$, be the vector of poles. We have two essentially different types of poles, finite and infinite. For the infinite poles, we add vectors $A^k v$ to our space and for the finite poles vectors $\left(\prod_{j=k,\sigma_j\neq\infty}^{1}(A-\sigma_j I)^{-1}\right)v$. For $\sigma = [\infty, \sigma_2, \sigma_3, \infty, \ldots]$ the rational Krylov subspace starts with

$$\mathcal{K}_m^{\text{rat}}(A, v, \sigma) = \left\{v, Av, (A-\sigma_2 I)^{-1}v, (A-\sigma_3 I)^{-1}(A-\sigma_2 I)^{-1}v, A^2 v, \ldots\right\}. \tag{2.1}$$

The shifts for finite poles provide additional flexibility, which is beneficial in some applications. For the infinite poles, we can also shift A and add $(A-\zeta_k)v$ instead, but this provides no additional flexibility, since the spanned space is not changed: Let $\mathcal{K}_m(A, v)$ be a standard Krylov subspace of dimension m as in (1.1). Then

$$\operatorname{span}\left\{\mathcal{K}_m(A, v) \cup \operatorname{span}\left\{A^m v\right\}\right\} = \operatorname{span}\left\{\mathcal{K}_m(A, v) \cup \operatorname{span}\left\{(A-\zeta_k I)^m v\right\}\right\}. \tag{2.2}$$

Let V span the rational Krylov subspace of dimension m such that

$$\operatorname{span}\left\{V_{:,1:k}\right\} = \mathcal{K}_k^{\text{rat}}(A, v, \sigma) \quad \forall k \leq m, \tag{2.3}$$

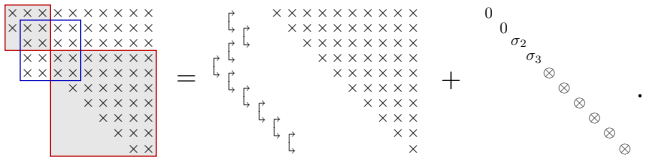
and let $H = V^*AV$. The matrix $H - D$, where D is a diagonal matrix with

$$D_{1,1} = 0 \qquad \text{and} \qquad D_{i,i} = \begin{cases} \sigma_{i-1}, & \sigma_{i-1} \neq \infty, \\ 0, & \sigma_{i-1} = \infty, \end{cases} \quad i = 2, \ldots, n-1, \tag{2.4}$$

is of extended Hessenberg structure, see [18, Section 2.2], [6]. If σ_i is an infinite pole, then the $(i-1)$st rotation is positioned on the left of the ith rotation. If, instead, σ_i is finite, then the $(i-1)$st rotator is on the right of the ith rotation.

For the Krylov subspace in (2.1), the matrix H has the structure

(2.5)



The matrix H consists of overlapping Hessenberg (first and last square) and inverse Hessenberg blocks (second square). For infinite poles we are free to choose any shift as (2.2) shows. These shifts are marked by $\otimes$ in the scheme above. For convenience we will choose these poles equal to the last finite one.

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

2.2. Algorithm. We will now describe how to obtain the structure shown in the example above. The algorithm consists of three steps:

- Construct a large Krylov subspace $\mathcal{K}_{m+p}(A, v)$ spanned by the columns of V and set $H = V^*AV$.
- Transform, via unitary similarity transformations, the matrix H into the structure of the projected counterpart corresponding to the requested rational Krylov space.
- Retain only the upper left $m \times m$ part of H and the first m columns of V.

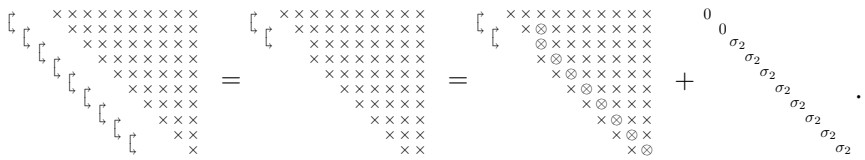
We will now explain these steps in detail by computing the rational Krylov subspace (2.1). The algorithm starts with a large Krylov subspace $\mathcal{K}_{m+p}(A, v)$. Let the columns of V span $\mathcal{K}_{m+p}(A, v)$ as in (1.2). Then the projection of A onto V yields a Hessenberg matrix $H = V^*AV$, that satisfies the equation

$$AV = VH + r \begin{bmatrix} 0 & 0 & \cdots & 1 \end{bmatrix}, \tag{2.6}$$

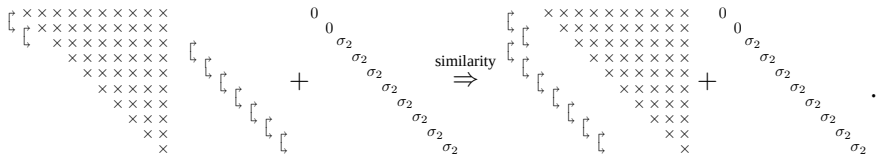
where r is the residual. The QR decomposition of H is computed and the Q factor is stored in the form of $n-1$ rotators. In case H is not unreduced, one has found an invariant subspace, often referred to as a lucky breakdown as the projected counterpart contains now all the essential information and one can solve the problem without approximation error; the residual becomes zero. Solving the resulting small dense problem in case of a breakdown is typically easy and will not be investigated here. Thus we assume that H is unreduced; hence all rotators in Q differ from the identity.

Let us now discuss the second bullet of the algorithm. The QR decomposition of the Hessenberg matrix $H = QR$ equals the left term in (2.7) and as an example we will, via unitary similarity transformations, bring it to the shape shown in (2.5). According to diagram (2.5) we keep the first two rotators but have to change the position of the third rotator. The third rotator is positioned on the right side of rotator two, which is wrong, thus we have to bring this rotator (and as we will see, including all the trailing rotators) to the other side. Therefore, we apply all rotators except the first two to R. Because of the descending ordering of the rotators, this creates new non-zero entries in the subdiagonal of R. We then introduce the pole σ_2: the diagonal matrix $\operatorname{diag}[0, 0, \sigma_2, \ldots, \sigma_2]$ is subtracted from QR. These steps are summarized in the following diagrams:

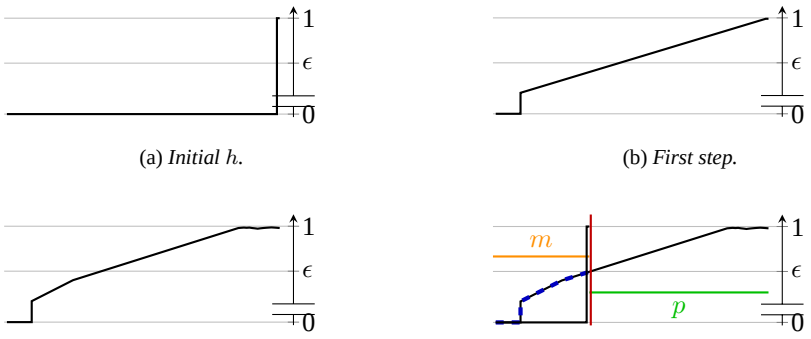
(2.7)



The elements marked by $\otimes$ are the ones that are changed by introducing the σ_2's on the diagonal. In the next step we restore the upper triangular matrix by applying rotators from the right. These rotations are then brought by a similarity transformation back to the left-hand side. This similarity transformation preserves the structure of D, as the same shift σ_2 appears in all positions in D from the third one on,



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

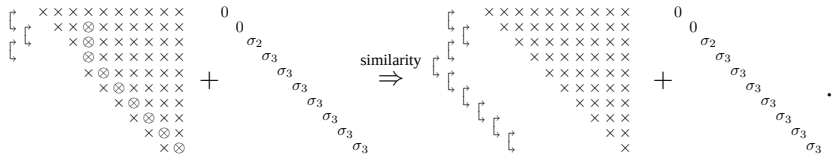


(a) *Initial* h. (b) *First step.*

(c) *After 15 steps.* (d) *Selecting the first vectors.*

FIG. 2.1. *Log-scale plots of the residual, showing the effect of the similarity transformation and the selection of the first vectors.*

The procedure is then repeated for all subsequent poles. The introduction of the second finite pole is illustrated in the following figures:



For the infinite poles, we do not change the pattern as we started from a matrix in Hessenberg form; we leave it like that. But, we do keep the possible non-zero shifts present on the diagonal matrix. We could try to change them and set them to zero, but this would require unnecessary computations and (2.2) shows that this is redundant.

These transformations bring H to the desired extended Hessenberg plus diagonal structure (2.5). But, considering (2.6) we see that the residual also gets affected, which is an undesired side-effect. The similarity transformations that we apply to H correspond to unitary matrices, which are applied from the right to (2.6). The residual matrix R is of rank 1 and initially has the following structure

$$R = rh = r \begin{bmatrix} 0 & 0 & \cdots & 1 \end{bmatrix}.$$

The first similarity transformation corresponding to a finite pole results in applying a series of rotators to h, thereby immediately destroying the zero pattern and resulting in a rather dense vector $\tilde{h}$. However, since the norm is preserved under unitary transformations, we observe that the energy of h gets distributed over many components in $\tilde{h}$; the absolute values of the entries in $\tilde{h}$ are typically decaying from $\tilde{h}_n$ to $\tilde{h}_1$. This is sketched in Figure 2.1(a) and 2.1(b), where a logarithmic y-axis with an added point for 0 is used. The ϵ stands for the machine precision. Every time a similarity transformation linked to a finite pole is handled the "energy" is pushed a bit more to the left; see Figure 2.1(c). Finally we retain the first part of V, where the residual is often very small; see Figure 2.1(d).

We choose an oversampling parameter p that determines how many additional vectors we add to the standard Krylov subspace. Since we keep m vectors, we start with $m + p$ ones. By applying the similarity transformations, we change V, H, and h in (2.6). At the end, we select the leading $m \times m$ block of H. The approximation is successful if the entries of the new residual (blue dashed part in Figure 2.1(d)) are sufficiently small, as in this case we have numerically computed the projected counterpart linked to the rational Krylov space. This will be shown by the implicit-Q-theorem in the next subsection.

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

2.3. The implicit-Q-theorem. The following variant of the implicit-Q-theorem in [17] shows that the algorithm described in the last subsection leads indeed to an approximation of the rational Krylov subspace sought after. It is shown that there is essentially one extended Hessenberg plus diagonal matrix with the prescribed structure, which is at the same time the projection of A onto the range of V, with $Ve_1 = v$.

THEOREM 2.1. *Let A be a regular*[3] *$n \times n$ matrix, σ and $\hat{\sigma}$ be two shift vectors, and let $\underline{V}$ and $\underline{\hat{V}}$ be two $n \times (m+1)$ rectangular matrices having orthonormal columns, sharing the first column $\underline{V}e_1 = \underline{\hat{V}}e_1$. Let V and $\hat{V}$ consist of the first m columns of $\underline{V}$ and $\underline{\hat{V}}$, respectively. Consider*

$$AV = VH + rw_k^* = \underline{V}\,\underline{H} = \underline{V}\big(QR + D\big),$$
$$A\hat{V} = \hat{V}\hat{H} + \hat{r}\hat{w}_k^* = \underline{\hat{V}}\,\underline{\hat{H}} = \underline{\hat{V}}\big(\hat{Q}\hat{R} + \hat{D}\big),$$

where Q and $\hat{Q}$ are decomposed into a series of rotations, denoted by G_i^Q and $G_i^{\hat{Q}}$, and ordered as imposed by σ and $\hat{\sigma}$. Let further $H - D$ and $\hat{H} - \hat{D}$ be invertible.

Then define $\hat{k}$ as the minimum

$$\hat{k} = \min_i \left\{ 1 \le i \le n-2 \text{ such that, } G_i^Q = I, G_i^{\hat{Q}} = I, \text{ or } \sigma_{i-1} \ne \hat{\sigma}_{i-1} \right\},$$

if no such $\hat{k}$ exists, set it equal to m.

Then the first $\hat{k}$ columns of V and $\hat{V}$, and the upper left $\hat{k} \times \hat{k}$ blocks of V^*AV and $\hat{V}^*A\hat{V}$ are essentially the same, meaning that there is a diagonal matrix E, with $|E_{i,i}| = 1$, such that $VE = \hat{V}$ and $E^*V^*AVE = \hat{V}^*A\hat{V}$.

To prove this theorem the following lemma is required, which is the rational Krylov analog of [28, Theorem 3.7].

LEMMA 2.2. *Let H be an $n \times n$ matrix, with*

$$H = QR + D,$$

where Q is unitary with a decomposition into rotations according to a shift vector σ, R an upper triangular matrix, and D a diagonal matrix containing the poles as in (2.4). Let further $H - D$ be unreduced. Then for $k = 1, \ldots, n-1$,

$$\operatorname{span}\{e_1, \ldots, e_k\} = \mathcal{E}_k = \mathcal{K}_k^{rat}(H, e_1, \sigma).$$

Proof. First we show as in [28, Lemma 3.6] that for $k = 1, \ldots, n-2$,

(a) if $\sigma_k = \infty$, then $H\mathcal{K}_k^{\text{rat}}(H, v, \sigma) \subseteq \mathcal{K}_{k+1}^{\text{rat}}(H, v, \sigma)$ and

(b) if $\sigma_k \ne \infty$, then $(H - \sigma_k I)^{-1}\mathcal{K}_k^{\text{rat}}(H, v, \sigma) \subseteq \mathcal{K}_{k+1}^{\text{rat}}(H, v, \sigma)$.

Let

$$\mathcal{K}_k^{\text{rat}}(H, v, \sigma) = \operatorname{span}\left\{ \left(\prod_{j=k, \sigma_j \ne \infty}^{1} (H - \sigma_j I)^{-1} \right) v, \ldots, v, \ldots, H^{q_k} v \right\},$$

with $q_k = |\{i \le k | \sigma_i = \infty\}|$. Further let u_p be defined for $p \le k - q_k$ by

$$u_p := \left(\prod_{j=p, \sigma_j \ne \infty}^{1} (H - \sigma_j I)^{-1} \right) v,$$

[3] Regular in this case means invertible.

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

$p_- := \operatorname{argmax}_{i<p} \sigma_i \neq \infty$, and $p_+ := \operatorname{argmin}_{i>p} \sigma_i \neq \infty$.
If $\sigma_k = \infty$, then $HH^q v = H^{q+1} v$ and $Hu_p = (H - \sigma_{p_-} I)u_p + \sigma_{p_-} u_p \in \operatorname{span}\{u_{p_-}, u_p\}$.
If $\sigma_k \neq \infty$, then

$$\begin{aligned}(H - \sigma_k I)^{-1} H^q v &= (H - \sigma_k I)^{-1}(H - \sigma_k I + \sigma_k I)H^{q-1} \\ &= H^{q-1} v + \sigma_k (H - \sigma_k I)^{-1} H^{q-1} v\end{aligned}$$

and

$$\begin{aligned}(H - \sigma_k I)^{-1} u_p &= (H - \sigma_k I)^{-1}(H - \sigma_{p_+} I)(H - \sigma_{p_+} I)^{-1} u_p. \\ &= u_{p+1} + (\sigma_k - \sigma_{p_+})(H - \sigma_k I)^{-1} u_{p+1}.\end{aligned}$$

Let us now prove the lemma using the same argument as in [28, Theorem 3.7], i.e., by induction. The statement is obviously true for $k = 1$. We choose a decomposition of H of the form

$$H = G_L G_k G_R R + D,$$

where G_L and G_R are the rotators to the left and right of G_k respectively, the rotation acting on rows k and $k+1$.

Suppose that $\sigma_k = \infty$. Using (a) with $v = e_j$, $j \leq k$ shows that $H\mathcal{E}_k \subseteq \mathcal{K}^{\text{rat}}_{s,k}(H, e_1, \sigma)$. We will now show that there is an $x \in \mathcal{E}_k$ such that $z = Hx \in \mathcal{E}_{k+1}$ and $e^*_{k+1} z \neq 0$.

We set $x := R^{-1} G_R^{-1} e_k$. Since G_k is not in G_R and R is a regular upper triangular matrix $x \in \mathcal{E}_k$. The vector $y := G_k G_R R x$ is in $\mathcal{E}_{k+1}$ and since $G_k \neq I$, we have $e^*_{k+1} y \neq 0$. Further $G_L y \in \mathcal{E}_{k+1}$ since G_{k+1} is not in G_L because of $s_k = \ell$. The vector z defined by

$$z = (G_L G_k G_R R + D)x$$

has the desired structure since D is diagonal with $D_{k+1,k+1} = 0$.

We now suppose that $\sigma_k \neq \infty$. Let $y \in \operatorname{span}\{e_k, e_{k+1}\}$ be the solution of $G_k y = e_k$. Since $G_k \neq I$ we have $e^*_{k+1} y \neq 0$. We further have that $G_L e_k \in \mathcal{E}_k$ since $s_k = r$. We set $z := R^{-1} G_R^{-1} y \in \mathcal{E}_{k+1}$, with $e^*_{k+1} z \neq 0$ since R^{-1} is invertible. The vector $x := (G_L G_k G_R R + D - \sigma_k I)z$ is in $\mathcal{E}_k$ since $D - \sigma_k I$ is a diagonal matrix with $(D - \sigma_k I)_{k+1,k+1} = 0$. Thus, we have a pair (x, z) with $z = (H - \sigma_k I)^{-1} x$. This completes the proof. ∎

Proof of Theorem 2.1. The proof is a partial analog of [17, Theorem 3.5]. Let us now assume that $\sigma = \hat{\sigma}$. Let further $K^{\text{rat}}_n(H, e_1, \sigma)$ be the Krylov matrix having as columns the vectors iteratively constructed for generating the associated Krylov subspace $\mathcal{K}^{\text{rat}}_n(H, e_1, \sigma)$. Then we know from Lemma 2.2 that $K^{\text{rat}}_n(H, e_1, \sigma)$ is upper triangular. Since it holds that

$$\begin{aligned}&VK^{\text{rat}}_n(H, e_1, \sigma) = K^{\text{rat}}_n(VHV^*, Ve_1, \sigma) = K^{\text{rat}}_n(A, Ve_1, \sigma) = \\ &K^{\text{rat}}_n(A, \hat{V}e_1, \sigma) = K^{\text{rat}}_n(\hat{V}\hat{H}\hat{V}^*, \hat{V}e_1, \sigma) = \hat{V}K^{\text{rat}}_n(\hat{H}, e_1, \sigma),\end{aligned}$$

$VK^{\text{rat}}_n(H, e_1, \sigma)$ and $\hat{V}K^{\text{rat}}_n(\hat{H}, e_1, \sigma)$ are QR decompositions of the same matrix and thus V and $\hat{V}$, and H and $\hat{H}$, are essentially the same for the full-dimensional case with identical shift vectors. By multiplication with $P_{\hat{k}} = [e_1, \ldots, e_{\hat{k}}]$ from the right, the equality can be restricted to the first $\hat{k}$ columns and the upper left $\hat{k} \times \hat{k}$ block. For the case $\sigma \neq \hat{\sigma}$ and if one of the matrices is not unreduced, we refer to the proof of [17, Theorem 3.5]. ∎

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

2.4. A numerical example. For this and all other numerical experiments in this paper, we use MATLAB implementations of the algorithms. In the (block) rational cases reorthogonalization has been used when generating the orthogonal bases. The experiments have been performed on an Intel® Core™i5-3570 (3.40GHz). The following example is an extension of [17, Example 6.5].

EXAMPLE 2.3. We choose $A \in \mathbb{R}^{200\times 200}$ to be a diagonal matrix with equidistant eigenvalues $\{0.01, 0.02, \ldots, 2\}$. We used the approximate rational Krylov subspace $\mathcal{K}_m^{\text{rat}}(A, v, \sigma)$ to approximate $f(A)v$ as

$$f(A)v \approx Vf(H)V^*v = Vf(H)e_1 \|v\|_2 ,$$

with the columns of $V_{:,1:j}$ spanning $\mathcal{K}_j^{\text{rat}}(A, v, \sigma)$ for all $j \le m$ and $H = V^*AV$. The entries of the vector v are normally distributed random values with mean 0 and variance 1. To demonstrate the power of shifts, we choose a continuous function $f_{[0.10,0.16]}$ focusing on a small part of the spectrum:

$$f_{[0.10,0.16]}(x) = \begin{cases} \exp(-100\,(0.10 - x)), & x < 0.10, \\ 1, & x \in [0.10, 0.16], \\ \exp(-100\,(x - 0.16)), & x > 0.16. \end{cases}$$

In Figure 2.2 we compare three different Krylov subspaces. The green line shows the accuracy of the approximation of $f_{[0.10,0.16]}(A)v$ with $\mathcal{K}_m(A, v)$, the red line is based on the approximate extended Krylov subspace $\mathcal{K}_m^{\text{rat}}(A, v, [0, \infty, 0, \infty, \ldots])$, and the orange line links to $\mathcal{K}_m^{\text{rat}}(A, v, [0.115, \infty, 0.135, \infty, 0.155, \infty, 0.105, \infty, \ldots])$ computed as an approximate rational Krylov subspace. For the latter two subspaces we use the algorithm described in Subsection 2.2, where we have chosen the oversampling parameter $p = 100$. In Figure 2.3 we compare the approximate rational Krylov subspaces for different oversampling parameters p. The approximate rational Krylov subspaces are computed from larger Krylov subspaces and thus their accuracy cannot be better. The gray lines show the expected accuracy based on the large Krylov subspace.

The use of the shifts $(0.115, 0.135, 0.155, 0.105)$ improves the accuracy significantly. The shifts boost the convergence on the relevant interval $[0.10, 0.16]$. This can also be observed in the plots of the Ritz values in Figure 2.4. In Figure 2.4(a) the Ritz values for the standard Krylov subspace are plotted. Each column in this plot shows the Ritz value of one type of subspace for dimensions 1 to 160. Red crosses stand for Ritz values approximating eigenvalues with an absolute error smaller than $10^{-7.5}$; orange crosses indicate good approximations with absolute errors between $10^{-7.5}$ and 10^{-5}; the green crosses are not so good approximations with errors between 10^{-5} and $10^{-2.5}$. The typical convergence behavior to the extreme eigenvalues is observed.

Figure 2.4(b) shows the Ritz values of the approximate rational Krylov subspaces computed with our algorithm and the above mentioned shifts. One can clearly see that well-chosen shifts ensure that the relevant information moves to the first vectors. In and nearby $[0.10, 0.16]$, there are only tiny differences compared with Figure 2.4(c), where we see the Ritz values obtained with the exact rational Krylov subspace.

Finally, Figure 2.4(d) shows the Ritz values determined with the exact extended Krylov subspace. The Ritz values in $[0.10, 0.16]$ approximate the eigenvalues much later than in the previous plot and, thus, the accuracy of the approximation of $f_{[0.10,0.16]}(A)v$ by an approximate, extended Krylov subspace, red graph in Figure 2.2, is not as good as for the rational Krylov subspace, orange graph.

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

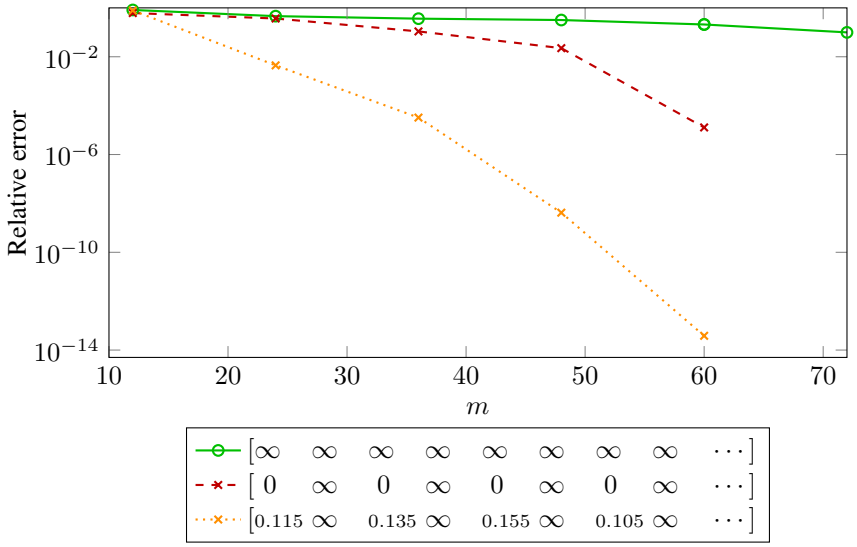


FIG. 2.2. *Relative error in approximating* $f_{[0.10,0.16]}(A)v$ *for* $m = 12, 24, 36, 48, 60$, $p = 100$.

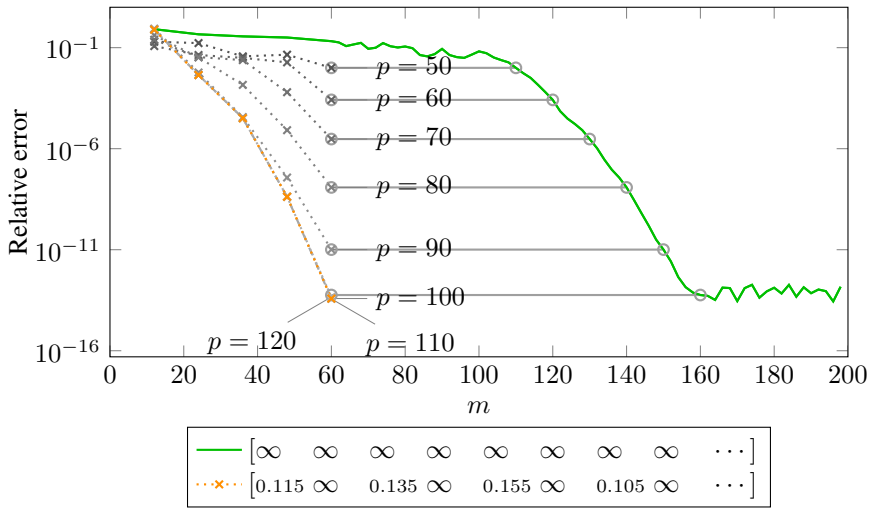


FIG. 2.3. *Relative error in approximating* $f_{[0.10,0.16]}(A)v$.

The first three plots of Figure 2.4 have been merged into a video[4] allowing easy comparison.

3. Block Krylov subspaces. Computing $f(A)v_1, \ldots, f(A)v_b$ simultaneously can be done by a block Krylov subspace of the form

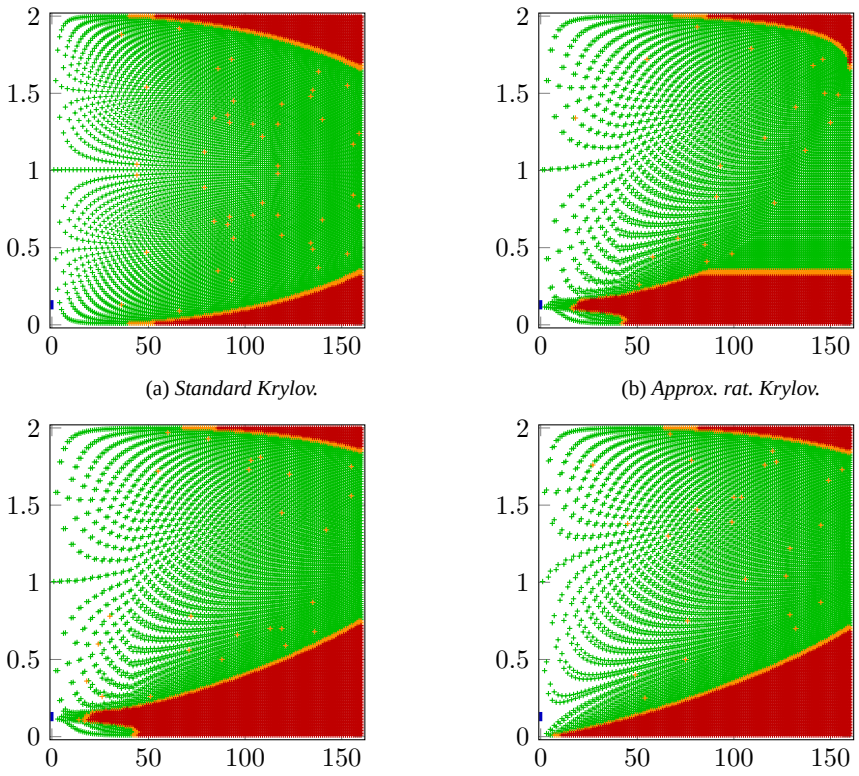
$$\mathcal{K}_m(A, \mathcal{V}) = \operatorname{span}\left\{\mathcal{V}, A\mathcal{V}, A^2\mathcal{V}, A^3\mathcal{V}, \ldots, A^{m/b-1}\mathcal{V}\right\} \quad \text{with} \quad \mathcal{V} = [v_1, \ldots, v_b]\,.$$

The dimension of $\mathcal{K}_m(A, \mathcal{V})$ is m and must be an integer multiple of b.

We will first analyze the structure of the matrix H, the projection of A onto the Krylov

[4] http://etna.math.kent.edu/vol.43.2014/pp100-124.dir/rational_eq_spaced.mp4

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



(a) *Standard Krylov.* (b) *Approx. rat. Krylov.*

(c) *Rational Krylov.* (d) *Extended Krylov.*

FIG. 2.4. *Ritz value plots for equidistant eigenvalues in* $[0, 2]$*; the interval* $[0.10, 0.16]$ *is marked blue.*

subspace $\mathcal{K}_k^{\text{rat}}(A, \mathcal{V}, \sigma)$, before we explain the necessary transformations to achieve this structure.

3.1. The structure of the projected counterpart for block Krylov subspaces. Let $\mathcal{V}$ be a tall and skinny matrix containing the starting vectors, $\mathcal{V} = [v_1, \ldots, v_b] \in \mathbb{C}^{n\times b}$, where b is the block-size. The rational Krylov subspace contains positive powers of A, $A^i\mathcal{V}$, for $\sigma_i = \infty$, and negative powers[5], $\left(\prod_{t=i,\sigma_t\neq\infty}^{1}(A-\sigma_t I)^{-1}\right)\mathcal{V}$, for $\sigma_i \neq \infty$.

Let $K := K_n^{\text{rat}}(A, \mathcal{V}, \sigma) \in \mathbb{C}^{n\times n}$ be the Krylov matrix linked to $\mathcal{K}_n^{\text{rat}}(A, \mathcal{V}, \sigma)$. The columns of K are the vectors of $\mathcal{K}_n^{\text{rat}}(A, \mathcal{V}, \sigma)$ without orthogonalization, while the columns of V, defined as in (2.3), form an orthonormal basis of this Krylov subspace. We assume that for all $i \in \{1, \ldots, b\}$ the smallest invariant subspace of A containing v_i is $\mathbb{C}^n$. Then there is an invertible, upper triangular matrix U, so that $K = VU$. Since the Krylov subspace is of full dimension, we have $AV = VH$ and $AKU^{-1} = KU^{-1}H$. Setting $H_K := U^{-1}HU$ yields

$$AK = KH_K. \tag{3.1}$$

Since U and U^{-1} are upper triangular matrices the QR decomposition of H has the same pattern of rotators as H_K. We will derive the structure of H based on the structure of H_K.

3.1.1. The structure of the projected counterpart for rational Krylov subspaces spanned by a non-orthogonal basis. We describe the structure of H_K and show that the

[5] $\prod_{t=i,\sigma_t\neq\infty}^{1}(A-\sigma_t I)^{-1}$ denotes the product is $(A-\sigma_t I)^{-1}\cdots(A-\sigma_1 I)^{-1}$.

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

QR decomposition of $H_K - D = Q\tilde{R}$, where D is a diagonal matrix based on the shifts, has a structured pattern of rotators. The following example will be used to illustrate the line of arguments, $\sigma = [\infty, \sigma_2, \sigma_3, \infty, \sigma_5, \infty, \infty, \dots]$. The corresponding Krylov matrix K is

$$K_n^{\text{rat}}(A, \mathcal{V}, \sigma) = \Big[\mathcal{V}, A\mathcal{V}, (A-\sigma_2 I)^{-1}\mathcal{V}, (A-\sigma_3)^{-1}(A-\sigma_2 I)^{-1}\mathcal{V}, A^2\mathcal{V}, \\ (A-\sigma_5 I)^{-1}(A-\sigma_3)^{-1}(A-\sigma_2 I)^{-1}\mathcal{V}, A^3\mathcal{V}, A^4\mathcal{V}\dots\Big]. \tag{3.2}$$

Inserting (3.2) into (3.1) provides

$$K_n^{\text{rat}}(A, \mathcal{V}, \sigma)H_K = \Big[A\mathcal{V}, A^2\mathcal{V}, A(A-\sigma_2 I)^{-1}\mathcal{V}, A(A-\sigma_3)^{-1}(A-\sigma_2 I)^{-1}\mathcal{V}, \\ A^3\mathcal{V}, A(A-\sigma_5 I)^{-1}(A-\sigma_3)^{-1}(A-\sigma_2 I)^{-1}\mathcal{V}, A^4\mathcal{V}, A^5\mathcal{V}\dots\Big]. \tag{3.3}$$

The matrix H_K consists of blocks of size $b \times b$. We will now show that H_K in the example (3.3) satisfies

$$H_K := \begin{bmatrix} 0 & 0 & I & 0 & 0 & 0 & 0 & \dots \\ I & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\ & 0 & \sigma_2 I & I & 0 & 0 & 0 & \dots \\ & 0 & 0 & \sigma_3 I & 0 & I & 0 & \dots \\ & I & 0 & 0 & 0 & 0 & 0 & \dots \\ & & & & 0 & \sigma_5 I & 0 & \dots \\ & & & & I & 0 & 0 & \dots \\ & & & & & & I & \dots \end{bmatrix}.$$

One can show that for $\sigma_j \neq \infty$,

$$A(A-\sigma_j I)^{-1} \prod_{\substack{t=j-1 \\ \sigma_t \neq \infty}}^{1} (A-\sigma_t I)^{-1}\,\mathcal{V} = \sigma_j \prod_{\substack{t=j \\ \sigma_t \neq \infty}}^{1} (A-\sigma_t I)^{-1}\,\mathcal{V} + \prod_{\substack{t=j-1 \\ \sigma_t \neq \infty}}^{1} (A-\sigma_t I)^{-1}\,\mathcal{V}.$$

Thus, from (3.3) it follows that the diagonal of H_K is D, where D is a diagonal matrix containing the shifts, cf. (2.4),

$$D = \text{blockdiag}\,(0I_b, \chi_1 I_b, \dots, \chi_{n-1} I_b) \quad \text{with} \quad \chi_i = \begin{cases} \sigma_i, & \sigma_i \neq \infty, \\ 0, & \sigma_i = \infty. \end{cases} \tag{3.4}$$

Let i and j be the indices of two neighboring finite shifts σ_i and σ_j, with $i < j$ and $\sigma_k = \infty$ $\forall i < k < j$. Then $H_K(bi+1 : b(i+1), bj+1 : b(j+1)) = I$. Additionally, for j, the index of the first finite shift, we have $H_K(1 : b, bj+1 : b(j+1)) = I$.

Let q be the index of an infinite shift. Then the associated columns of K and AK are

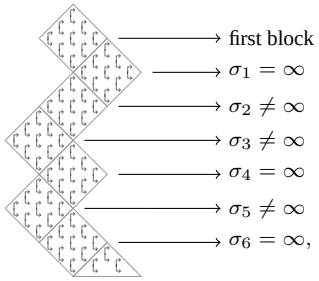
$$K_{:,bq:b(q+1)-1} = A^q\mathcal{V} \qquad \text{and} \qquad AK_{:,bq:b(q+1)-1} = A^{q+1}\mathcal{V}.$$

Thus, for two neighboring infinite shifts $\sigma_i = \infty$ and $\sigma_j = \infty$, with $i < j$ and $\sigma_k \neq \infty$ $\forall i < k < j$, we have $H_K(bj+1 : b(j+1), bi+1 : b(i+1)) = I$. Additionally, for j, the index of the first infinite, shift we have $H_K(bj+1 : b(j+1), 1 : b) = I$.

The column of H_K corresponding to the last infinite pole has a special structure related to the characteristic polynomial of A. For simplicity, we assume that the last shift is infinite and that the last block column of H_K is arbitrary. The matrix H_K is now completely determined.

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

In the next step, we compute the QR decomposition of H_K. For simplicity, we start with examining the case when all poles equal zero. Let us call this matrix H_K^0, with the QR decomposition $H_K^0 = Q_0 R_0$. The rhombi in Q_0 are ordered according to the shift vector σ. For the infinite shifts the rhombus is positioned on the right of the previous rhombus and for finite shifts on the left. Thus, e.g.,

$Q_0 =$ 

where all rotators in the rhombi are $\left[\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right]$, and

$$R_0 = \begin{bmatrix} I & & & \times \\ & \ddots & & \vdots \\ & & I & \times \\ & & & \diagdown \end{bmatrix},$$

where $\times$ now depicts a matrix of size $b \times b$ instead of a scalar. The rotations in the trailing triangle of Q_0 introduce the zeros in the last block column of R_0.

Let us now reconsider the rational case with arbitrary finite shifts. Let D be the diagonal matrix defined in (3.4). We then have $H_K - D = H_K^0 = Q_0 R_0$.

3.1.2. The structure of the projected counterpart for rational Krylov subspaces spanned by an orthogonal basis. We use the QR decomposition $H_K - D = Q_0 R_0$ to compute the QR decomposition of H. The matrix H can be expressed as

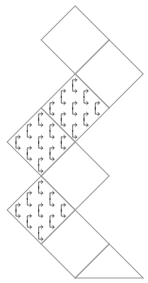
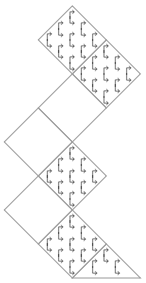
$$H = U H_K U^{-1} = U \left(Q_0 R_0 U^{-1} + D U^{-1} - U^{-1} D \right) + D,$$

since $D - U U^{-1} D = 0$. The matrix $W = D U^{-1} - U^{-1} D$ is upper triangular. If $\sigma_i = \infty$, then $D_{\rho(i),\rho(i)} = 0$, with $\rho(i)$ the set of indices $\{bi+1, bi+2, \ldots, bi+b\}$ for $i \geq 0$. Thus, if $\sigma_i = \infty$ and $\sigma_j = \infty$, then $W_{\rho(i),\rho(j)} = 0$. Further, $W_{\rho(i),\rho(i)} = 0$ since $D_{\rho(i),\rho(i)} = \sigma_i I$; see (3.4). In the example (3.1), W is a block matrix with blocks of size $b \times b$ and the following sparsity structure:

$$W = \begin{bmatrix} 0 & 0 & \times & \times & 0 & \times & 0 & 0 \\ & 0 & \times & \times & 0 & \times & 0 & 0 \\ & & 0 & \times & \times & \times & \times & \times \\ & & & 0 & \times & \times & \times & \times \\ & & & & 0 & \times & 0 & 0 \\ & & & & & 0 & \times & \times \\ & & & & & & 0 & 0 \\ & & & & & & & 0 \end{bmatrix}.$$

We now factor Q_0 as $Q_0^r Q_0^\ell$, where all blocks, which are on the left of their predecessor, are put into Q_0^r and the others into Q_0^ℓ,

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

$Q_0^r =$  , $Q_0^\ell =$  .

Since Q_0^ℓ consist solely of descending sequences of rhombi, the matrix $H_\ell = Q_0^\ell R_0 U^{-1}$ is of block Hessenberg form, in this example:

$$H_\ell = \begin{bmatrix} 0 & 0 & I & & & & & \times \\ I & 0 & 0 & & & & & \times \\ & I & 0 & & & & & \times \\ & & & I & & & & \times \\ & & & & 0 & I & & \times \\ & & & & I & 0 & & \times \\ & & & & & & 0 & \times \\ & & & & & & I & \times \end{bmatrix}.$$

Recall that we can write H as

$$H = U\left(Q_0^r H_\ell + DU^{-1} - U^{-1}D\right) + D = UQ_0^r\left(H_\ell + Q_0^{r*}W\right) + D.$$

Since W is a block upper triangular matrix with zero block diagonal and Q_0^{r*} contains only descending sequences of rhombi, the product $Q_0^{r*}W$ is block upper triangular, in this example:

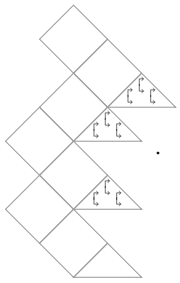
$$Q_0^{r*}W = \begin{bmatrix} 0 & 0 & \times & \times & 0 & \times & 0 & 0 \\ & 0 & \times & \times & 0 & \times & 0 & 0 \\ & & 0 & 0 & 0 & \times & 0 & 0 \\ & & & \times & \times & \times & \times & \times \\ & & & & \times & \times & \times & \times \\ & & & & & 0 & 0 & 0 \\ & & & & & & \times & \times \\ & & & & & & & 0 \end{bmatrix}.$$

For $\sigma_i \neq \infty$ we get a non-zero block $(Q_0^{r*}W)_{\rho(i+1),\rho(i+1)}$, since for each $\sigma_i \neq \infty$ the block rows $\rho(i)$ and $\rho(i+1)$ are swapped. However, since $W_{\rho(i+1),\rho(i)} = 0$ the block $(Q_0^{r*}W)_{\rho(i),\rho(i)}$ is zero if additionally $\sigma_{i-1} = \infty$. Hence, the sum of H_ℓ and $Q_0^{r*}W$ is also block Hessenberg with the same block subdiagonal as H_ℓ. In this example the sum is

$$H_\ell + Q_0^{r*}W = \begin{bmatrix} 0 & 0 & \otimes & \times & 0 & \times & 0 & \times \\ I & 0 & \times & \times & 0 & \times & 0 & \times \\ & I & 0 & 0 & 0 & \times & 0 & \times \\ & & & \otimes & \times & \times & \times & \times \\ & & & & \times & \otimes & \times & \times \\ & & & & I & 0 & 0 & \times \\ & & & & & & \times & \times \\ & & & & & & I & \times \end{bmatrix}.$$

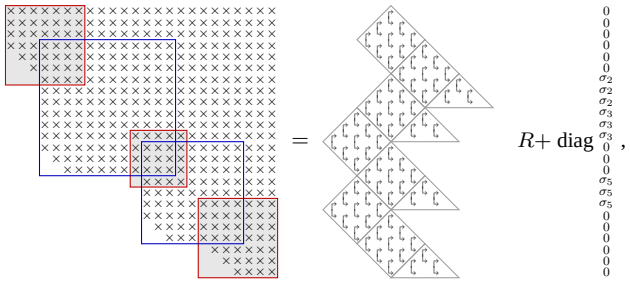
ETNA
Kent State University
http://etna.math.kent.edu

We now determine $Q_1 = Q_0^r Q_1^\ell Q_1^t$, where Q_1^ℓ and Q_0^ℓ have the same pattern of rotators and Q_1^t will be added later. The rotations in Q_1^ℓ have to be chosen so that $H_\ell + Q_0^{r*} W$ becomes block upper triangular and so that the blocks $\rho(i), \rho(i)$ with $\sigma_i = \infty$ or $i = 0$ also are upper triangular. Because of the special structure of $H_\ell + Q_0^{r*} W$ and Q_1^ℓ this is possible. The remaining blocks in this example can be brought into upper triangular form by the rotators in Q_1^t:

$Q_1^t =$  .

After passing Q_1 through the upper triangular matrix U to the right, we have the QR decomposition of $H - D$.

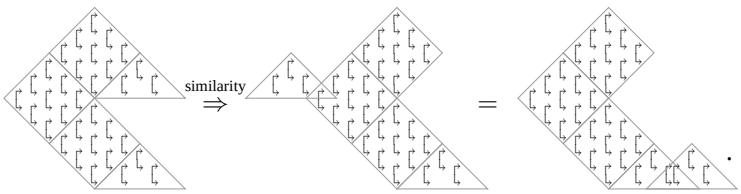
Summarizing the steps above, we have shown that the projection of A onto a block rational Krylov subspace such as (3.2) spanned by the matrix V leads to a structure of the form

$H = V^* A V =$  $R +$ diag ,

with R an upper triangular matrix.

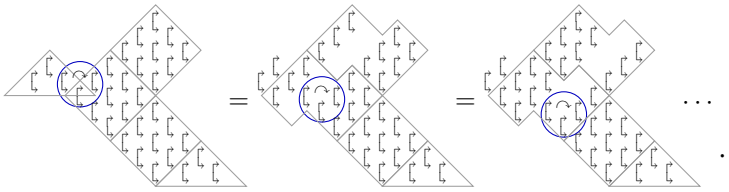
This structure is not suitable for our algorithm, since the QR decomposition of $H - D$ for the Krylov subspace with solely infinite poles does not have the additional rotators in Q_1^t. We will now show that there are similarity transformations that remove the rotators in Q_1^t. These transformations change the basis of the Krylov subspace but only within the block columns. Thus, the approximation properties are not affected if we always select full blocks.

The following three structure diagrams show the main steps:

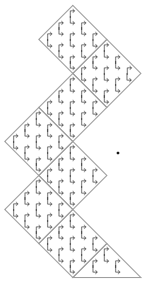
(3.5) 

First we bring the middle triangle to the other side. It has to be passed through the upper triangular matrix first and next a unitary similarity transformation eliminates the triangle on the right and reintroduces it on the left. This transformation only changes columns within one block. After that, a series of turnovers (blue circles) brings the rotators in the triangle down to the next triangle:

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

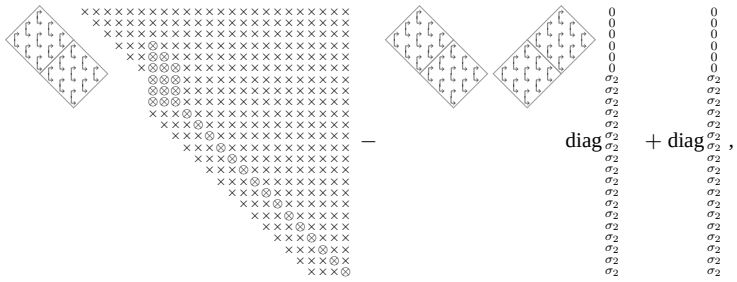
(3.6) 

Doing this for every rotation in the triangle completes (3.5). Finally, we can merge the two triangles; in this example with $b = 3$: fuse the rotations in the middle, do a turnover, and fuse the pairs on the left and right. Thus bringing H into a shape without the rotations in Q_1^t is sufficient to approximate the blocks of the block rational Krylov subspace. However, we are not able to approximate the individual vectors $\mathcal{K}_n^{\text{rat}}(A, \mathcal{V}, \sigma)$ and thus the Krylov condition that $V_{:,1:j}$ spans the first j vectors of $\mathcal{K}_n^{\text{rat}}(A, \mathcal{V}, \sigma)$ holds only for $j = ib$ with $i \in \mathbb{N}$. The desired shape in our example is:



3.2. The algorithm. We can now describe the algorithm to obtain the structure shown in the last subsection. The difference with respect to the algorithm from Subsection 2.2 is that now the rhombi instead of individual triangles are arranged according to the shift vector. For each $\sigma_i \neq \infty$, starting with $i = 1$, we have to introduce the pole and bring all the rhombi beginning with the $(i + 1)$st to the other side. After this has been done for the whole shift vector the first block columns are selected. The approximation is successful if the residual is small enough.

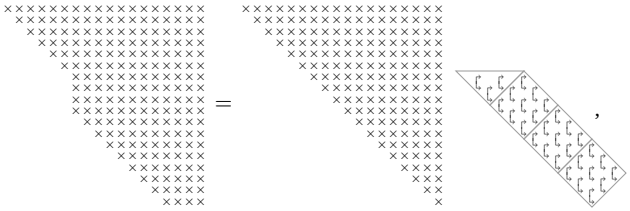
We will now describe in detail how to introduce one pole as this is the essential difference. If we apply the trailing rotations before introducing the shift, the matrix structure is not perturbed. Since the trailing rhombi form a descending sequence of rhombi, applying the rotations to the upper triangular matrix produces a Hessenberg matrix with b subdiagonals. Let $\sigma_2 \neq \infty$, and introduce the shift σ_2. The following diagram illustrates the introduction of the shift:



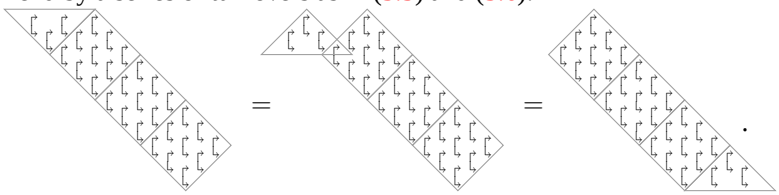
where the marked entries $\otimes$ represent the non-zero pattern of the second term. The transfer of the rotations is completed by pulling the rotators out to the right, thereby restoring the upper triangular shape. Unfortunately, this is not as simple as in the one-dimensional case with only

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

one vector. Because of the block structure, the zeroing of the entries based on rotators from the right-hand side leads to



where the rotations are not entirely in the desired pattern. We have to transform the V-pyramidal shape in the triangle into an A-pyramidal shape and then move the triangle to the lower end by a series of turnovers as in (3.5) and (3.6):



The rotations on the right-hand side of the upper triangular matrix are now in the right shape. We use a unitary similarity transformation to bring these rotators back to the left side of the matrix. Since this transformation must also be applied to the diagonal matrix containing the shifts, we have to use the same shift for all trailing positions as in Section 2. Then we continue with the next rhombus. If this rhombus corresponds to an infinite pole, nothing has to be done; also the shifts in D remain unaltered for convenience as in (2.5). If this rhombus corresponds to a finite pole, the trailing part of the matrix D is updated to the next shift. The process is continued until the desired shape is retrieved.

3.3. The implicit-Q-theorem. With the following theorem one can show that, in the absence of a residual, the above described algorithm computes a block rational Krylov subspace.

THEOREM 3.1. *Let A be a regular matrix, and let σ and $\hat{\sigma}$ be two shift vectors. Let $\underline{V}$ and $\underline{\hat{V}}$ be two $n \times (k+1)b$ rectangular matrices having orthonormal columns sharing the first b columns $\underline{V}[e_1, \ldots, e_b] = \underline{\hat{V}}[e_1, \ldots, e_b]$. Let V and $\hat{V}$ be the first kb columns of $\underline{V}$ and $\underline{\hat{V}}$, respectively. Consider*

$$AV = VH + rw_k^* = \underline{VH} = \underline{V}\big(QR + D\big),$$
$$A\hat{V} = \hat{V}\hat{H} + \hat{r}\hat{w}_k^* = \underline{\hat{V}\hat{H}} = \underline{\hat{V}}\big(\hat{Q}\hat{R} + \hat{D}\big),$$

where Q and $\hat{Q}$ are decomposed into a series of $b \times b$ rhombi of rotations ordered as imposed by σ and $\hat{\sigma}$ and let $H - D$ and $\hat{H} - \hat{D}$ be invertible.

Define $\hat{k}$ as the minimum index for which one of the $2b^2$ rotations in the ith rhombus of Q or $\hat{Q}$ is the identity or $\sigma_{i-1} \neq \hat{\sigma}_{i-1}$; if no such $\hat{k}$ exists, set it equal to $n-1$.

Then the first $\hat{k}b$ columns of V and $\hat{V}$, and the upper left $\hat{k} \times \hat{k}$ blocks of V^*AV and $\hat{V}^*A\hat{V}$ are block essentially the same. Block essentially the same *means here that $V_{:,jb+1:j(b+1)} = \hat{V}_{:,jb+1:j(b+1)}U$ with $U \in \mathbb{C}^{b\times b}$ and $U^*U = I$.*

The theorem is a generalization of Theorem 2.1 and can be shown analogously based on an analog generalization of Lemma 2.2 for the block case. Therefore, one has to show first that for $k = 1, \ldots, \frac{n}{b} - 2$,

(a) if $\sigma_k = \infty$, then $H\mathcal{K}_k^{\text{rat}}(H, \mathcal{V}, \sigma) \subseteq \mathcal{K}_{k+1}^{\text{rat}}(H, \mathcal{V}, \sigma)$ and
(b) if $\sigma_k \neq \infty$, then $(H - \sigma_k I)^{-1}\mathcal{K}_k^{\text{rat}}(H, \mathcal{V}, \sigma) \subseteq \mathcal{K}_{k+1}^{\text{rat}}(H, \mathcal{V}, \sigma)$.

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

The next step is to decompose H into

$$H = G_L G_k G_R R + D,$$

where G_k contains all rotators in the kth rhombus. Based on this decomposition, one can prove the block generalization of Lemma 2.2. As a block QR decomposition is block essentially unique we get

$$\begin{aligned} VK_n^{\text{rat}}(H, [e_1, \ldots, e_b], \sigma) = K_n^{\text{rat}}(VHV^*, V[e_1, \ldots, e_b], \sigma) = K_n^{\text{rat}}(A, \mathcal{V}, \sigma) = \\ K_n^{\text{rat}}(A, \mathcal{V}, \sigma) = K_n^{\text{rat}}(\hat{V}\hat{H}\hat{V}^*, \hat{V}[e_1, \ldots, e_b], \sigma) = \hat{V}K_n^{\text{rat}}(\hat{H}, [e_1, \ldots, e_b], \sigma). \end{aligned}$$

Thus, ensuring that the computed H has the desired structure is sufficient to compute an approximation to a block rational Krylov subspace, as illustrated by the numerical example in the next subsection.

3.4. A numerical example. The algorithm described above was used to approximately solve a Lyapunov equation

$$AX + XA + BB^* = 0,$$

for the unknown matrix X. The matrix $A \in \mathbb{R}^{5000\times 5000}$ is a diagonal matrix with entries

$$\lambda_i = 5.05 + 4.95\cos(\theta_i), \quad \theta_i \in [0, 2\pi) \quad \forall i,$$

having equally distributed θ_i. The matrix B is of size 5000×2, so that one actually needs a block Krylov algorithm with block-size $b = 2$. The dimension of B is the only point where this example differs from [16, Example 4.2]. The entries of B are computed with the MATLAB command `randn`, meaning they are pseudo-random based on a normal distribution with variance 1 and mean 0. A reference solution is computed with the MATLAB function `lyapchol`, which we assume to be exact. The approximate solution $\tilde{X} \approx X$ is computed based on the projection onto an approximate rational Krylov subspace via

$$\tilde{X} = VYV^*, \text{ where } Y \text{ is the solution of } HY + YH + (V^*B)(V^*B)^* = 0,$$

with $H = V^*AV$. In Figure 3.1, we compare the relative error for B of rank 2 (colored lines, bottom axis) with the results for a B of rank 1 (gray lines, larger marks, top axis). We need for the block-size $b = 2$ about twice as many vectors as for $b = 1$. The oversampling parameter p was chosen to be $100 \cdot b$. To make the comparison easier the gray lines are scaled according to the axis on top.

We observe that the relative accuracy shows almost the same behavior. According to the results from the last section, we also observe that the use of shifts (here $\{0.5, 0, 0.25, 0.125\}$ in round robin for the finite poles) improves the accuracy.

4. Symmetric matrices. If the matrix A is symmetric, then the Hessenberg matrix $H = V^*AV$ is also symmetric and thus tridiagonal. In this section we will exploit the symmetry when computing the approximate extended Krylov subspace. Therefore, we replace the QR decomposition of H by the LDL* factorization. Besides this adaption the algorithm remains the same and we can reuse the implicit-Q-theorem and the structure of H.

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

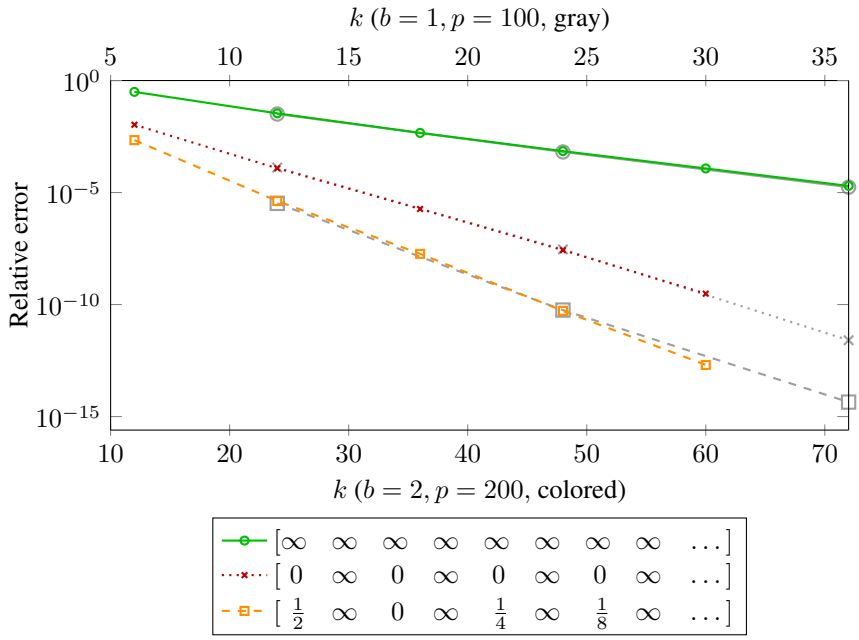
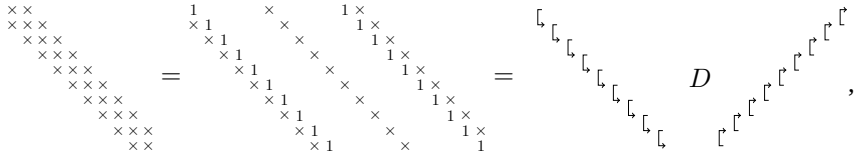


FIG. 3.1. *Relative error when solving* $AX + XA^* + BB^* = 0$ *for* $m = 12, 24, 36, 48, 60$.

4.1. Eliminators. The factorization $H = LDL^*$ of the tridiagonal matrix H is of the form



where we replaced the diagonal matrix in the middle by D to simplify the notation. In the second part of the equation, we factor the bidiagonal matrices into special Gaussian elimination matrices: identity matrices except for a single non-zero sub- (or super-)diagonal element. We use resembling symbols as for rotators and call them for short *eliminators*: $\left[\!\!\downarrow = \left[\begin{smallmatrix} 1 \\ \times & 1 \end{smallmatrix}\right]\right.$ is a *lower eliminator* and $\left[\!\!\uparrow = \left[\begin{smallmatrix} 1 & \times \\ & 1 \end{smallmatrix}\right]\right.$ an *upper* one. Applying such an eliminator to a matrix means adding a multiple of a row or column to another row or column respectively, e.g.,

$$\left[\!\!\downarrow \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} 1 & \\ \alpha & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} a \\ \alpha a + b \end{bmatrix} \quad \text{and also} \quad \left[\!\!\uparrow \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} 1 & \beta \\ & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} a + \beta b \\ b \end{bmatrix}.$$

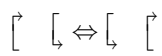
As with rotators we can perform several operations with eliminators. If the eliminators act on disjoint rows we can change their order arbitrarily. We can further change the ordering in the following cases:

$$\left[\!\!\uparrow \quad \left[\!\!\downarrow = \begin{bmatrix} 1 & \xi & \\ & 1 & \\ & & 1 \end{bmatrix} \begin{bmatrix} 1 & & \\ & 1 & \\ & \chi & 1 \end{bmatrix} = \begin{bmatrix} 1 & \xi & \\ & 1 & \\ & \chi & 1 \end{bmatrix} = \left[\!\!\downarrow \quad \left[\!\!\uparrow \quad \text{and} \quad \left[\!\!\downarrow \quad \left[\!\!\uparrow \quad = \left[\!\!\uparrow \quad \left[\!\!\downarrow .\right.\right.\right.\right.\right.\right.$$

We can also pass eliminators through diagonal matrices: applying an eliminator to a diagonal matrix creates a 1×1-bulge; this bulge can be removed by another eliminator acting from the other side. If we want to change the ordering of a lower and an upper eliminator acting on the

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

same rows,

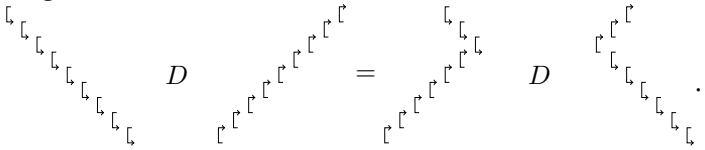


we have to compute the product and factor the resulting 2×2 matrix by an LDU or UDL factorization, respectively. This creates, however, an additional diagonal matrix. This diagonal matrix can be passed through the other eliminators and merged with the main diagonal.

4.2. The algorithm. We run the same algorithm, but replace the QR decomposition by the LDL* factorization and update this representation instead. First, we investigate the non-block, non-rational variant. We start with a large standard Krylov subspace with $H = LDL^*$.

The matrix H must now be brought into the shape determined by the shift vector. The lower eliminators on the left have the same pattern as the rotations in the general case. The pattern for the upper eliminators, however, is flipped. A similar structure has been used in [18, Section 4] to solve an inverse eigenvalue problem.

The transfer of rotators through the upper triangular matrix is replaced by bringing the corresponding lower eliminators from the left to the right and the corresponding upper eliminators from the right to the left. We start with the last eliminators. The trailing 2×2 matrix is given by an LDU factorization. We compute the UDL factorization of the product. The upper eliminator on the left commutes with the lower eliminators. Thus, the penultimate eliminators are now the most inner ones and we can repeat the process in the same way. After we have repeated this steps for the $n-4$ trailing eliminators, say for $\sigma = [\infty, 0, \infty, \dots]$, we have the following diagram:



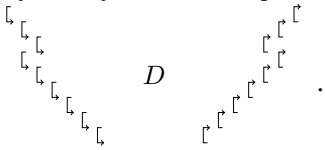
Now we have to bring the eliminators back to the other side. By a single unitary similarity transformation, we can bring both the lower eliminators on the right to the left and the upper eliminators on the left to the right. This can be achieved by

$$\begin{bmatrix} \overline{c} & -s \\ \overline{s} & c \end{bmatrix} \begin{bmatrix} 1 & \overline{\xi} \\ & 1 \end{bmatrix} = \begin{bmatrix} 1 & \\ \chi & 1 \end{bmatrix} \begin{bmatrix} d & \\ & e \end{bmatrix}, \qquad \begin{bmatrix} 1 & \\ \xi & 1 \end{bmatrix} \begin{bmatrix} c & s \\ -\overline{s} & \overline{c} \end{bmatrix} = \begin{bmatrix} \overline{d} & \\ & \overline{e} \end{bmatrix} \begin{bmatrix} 1 & \overline{\chi} \\ & 1 \end{bmatrix} \tag{4.1}$$

with

$$c = \frac{1}{\sqrt{1+|\xi|^2}}, \qquad s = \frac{\overline{\xi}}{\sqrt{1+|\xi|^2}}, \qquad d = \overline{c}, \qquad e = \overline{s\xi + d}, \quad \text{and} \qquad \chi = \xi.$$

This step also preserves the symmetry and we end up with an LDL* factorization of the form



After we have brought all the eliminators in the right shape determined by the shift vector, we select the first columns and the upper left part of H.

The main advantage of this eliminator-based approach is that we make full use of the symmetry. First of all that means that fewer unknown parameters in H, $\mathcal{O}(k+p)$ instead of $\mathcal{O}((k+p)^2)$, are needed. This reduces the number of required floating point operations. For

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

instance, passing one rotator through the upper triangular matrix changes $2(k+p)$ entries in the upper triangular matrix. By using the LDL* factorization we have to change only two entries on the diagonal. The reduced number of floating point operations also reduces the runtime of the algorithm; see Example 4.1. Unfortunately, the overall complexity is almost the same as for non-symmetric matrices. This will be illustrated and explained in the numerical example. Second, we preserve the symmetry and can exploit this in the remaining computations that have to be executed on the projected counterpart.

4.3. A numerical example. The matrices in [17, Examples 6.1–6.4] are all symmetric. The runtime of the symmetric variant is up to 5% less than the runtime of the non-symmetric implementation used in [17]. This small gain can be explained by the fact that the most expensive step, the update of the subspace V, which is of linear complexity in n, the dimension of A, is the same for the symmetric and the non-symmetric implementation. However, the accuracy of the symmetric variant is almost the same as we will see in the following example.

EXAMPLE 4.1. This example is identical to [13, Example 5], which has been used also in [17, Example 6.3] in the context of approximate extended Krylov subspaces without explicit inversion.

We compute the product of a matrix function and a vector, $f(A)v$, with $f(x) = 1/\sqrt{x}$, using an approximate, extended Krylov subspace. The matrix A is the discretization of the differential operator $L(u) = \frac{1}{10}u_{xx} - 100u_{yy}$ on the unit square. We use 40 equally spaced interior points. The discretization uses three-point stencils in both directions. Together with homogeneous boundary conditions the matrix A is symmetric, positive definite, and of size 1600×1600. The vector v is chosen to have the entries $v_j = 1/\sqrt{40}, \forall j$.

We choose the oversampling parameter p to be 200. In Figure 4.1 we can see almost no difference between the symmetric and the non-symmetric implementation; the crosses are always inside the circles. Thus the accuracy of the symmetric variant is as good as the one of the non-symmetric variant in this example.

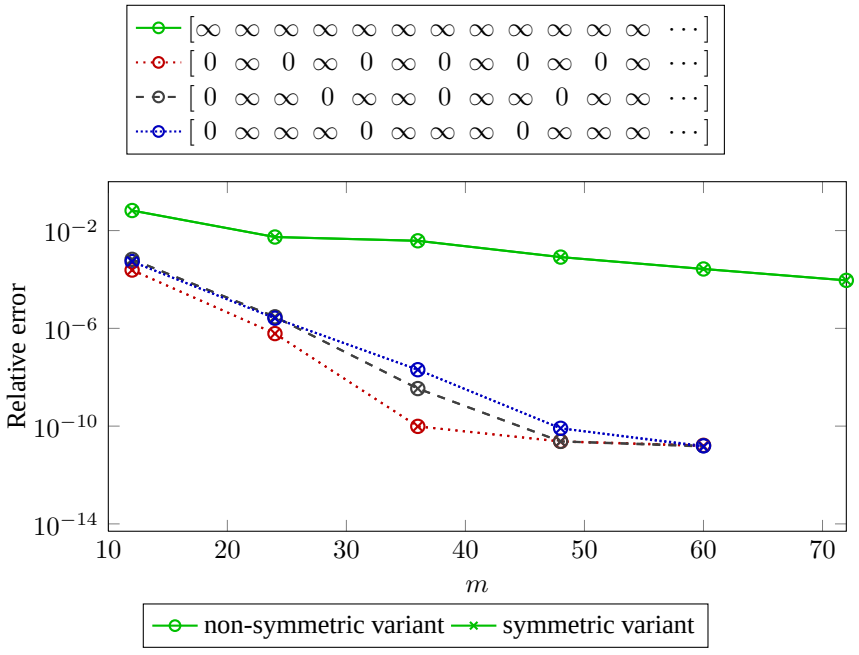
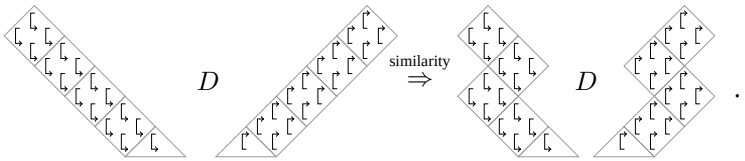


FIG. 4.1. *Relative error in approximating* $f(x) = 1/\sqrt{x}$ *for* $m = 12, 24, 36, 48, 60$.

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

4.4. Combination with block rational Krylov subspaces. Obviously one can combine the ideas for exploiting the symmetry with the algorithm for approximating a block rational Krylov subspace. This leads again only to a different implementation based on the more efficient representation of the symmetric matrix. Thus the theoretical results from Section 2 and 3 remain valid.

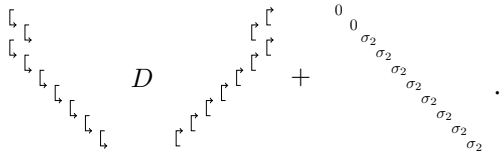
4.4.1. Block matrices. We will explain the block matrix approach for an example with $b = 2$, where we get a pentadiagonal matrix instead of the tridiagonal one as for $b = 1$. Hence the LDL* factorization of this matrix gives us two sequences of eliminators on both sides, which we can group in rhombi as in Section 3. Based on the shift vector similarity transformations are used to order the rhombi on both sides in a way that the result approximates a block extended Krylov subspace. For $\sigma = [\infty, 0, \infty]$ the following diagram sketches the shape:



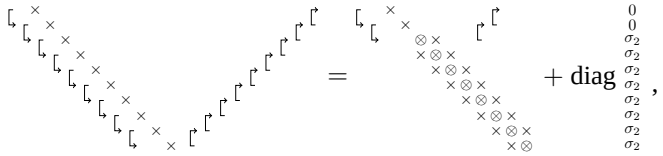
4.4.2. Rational Krylov subspaces. The LDL* factorization of the projected counterpart H of the rational Krylov subspace

$$\mathcal{K}^{\text{rat}}_{s,k}(A, v, \sigma) = \text{span}\left\{v, Av, (A - \sigma_2 I)^{-1}v, A^2v, A^3v, \ldots\right\}$$

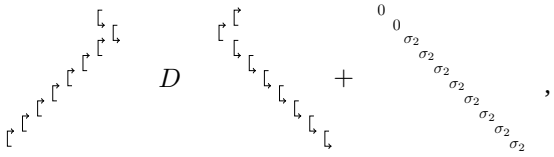
with symmetric A looks like



For the introduction of the shifts a similar trick as for the rational case is used: we apply the trailing eliminators to the diagonal matrix and get a tridiagonal matrix. Then the shifts are introduced and the tridiagonal matrix is refactored. The intermediate step is



where the entries $\otimes$ are changed by introducing the shifts. We observe that the diagonal matrix that is subtracted from the tridiagonal matrix is not changed by applying the inverses of the four eliminators. Next the UDU* factorization of the tridiagonal block is computed. Hence, we get (the diagonal matrix equals D)



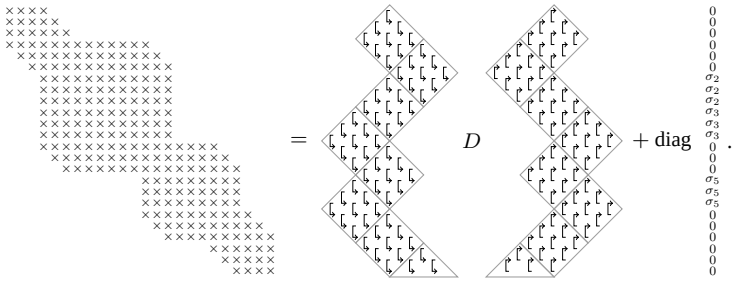
where we now can use rotations to bring the trailing eliminators simultaneously by unitary similarity transformations back to the other side as in (4.1). If the desired rational Krylov subspace has several finite poles the above described steps have to be repeated.

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

4.4.3. Block rational Krylov subspaces. We just provide an example pattern of a symmetric block rational Krylov subspace for $b = 3$. The necessary steps to achieve this pattern are analogous to the previous sections. The projected counterpart H of the block rational Krylov subspace

$$\mathcal{K}^{\text{rat}}_{[\ell r r \ell r \ell \ell],8}(A, \mathcal{V}, \sigma) = \operatorname{span}\left\{\mathcal{V}, A\mathcal{V}, (A - \sigma_2 I)^{-1}\mathcal{V}, \ldots\right\},$$

with $A = A^*$ and $\mathcal{V} \in \mathbb{C}^{n\times 3}$ has the factorization:



5. Conclusions. We have presented an algorithm to approximately compute rational Krylov subspaces and rational block Krylov subspaces. We explained how to exploit the symmetry of the original matrix. The numerical experiments illustrate that the algorithm is efficient for some of the examples. The algorithm can be interpreted as a compression algorithm operating on an oversampled large Krylov subspace, and this implies that it cannot add new data in the compression step. Unfortunately, this means that the algorithm fails to deliver good results for those applications or examples where the large Krylov subspace lacks the information on the inverse.

Even though this is a major step forward towards an algorithm of practical use, further research is necessary. Future investigations include preliminary analysis of the matrices to predict whether the algorithm will succeed, incorporating preconditioning, examining possible extensions to bi-orthogonal Krylov methods, and incorporation of good pole selection. When testing the algorithm on some rational Krylov spaces, we accidently picked poles equal to the eigenvalues, and even, though the associated Krylov space is ill-defined, the algorithm performed well. This behavior requires further study.

Acknowledgments. The authors thank the referees for their valuable comments.

REFERENCES

[1] A. C. ANTOULAS, *Approximation of Large-Scale Dynamical Systems*, SIAM, Philadelphia, 2005.
[2] T. BREITEN AND T. DAMM, *Krylov subspace methods for model order reduction of bilinear control systems*, Systems Control Lett., 59 (2010), pp. 443–450.
[3] T. DAMM, *Direct methods and ADI-preconditioned Krylov subspace methods for generalized Lyapunov equations*, Numer. Linear Algebra Appl., 15 (2008), pp. 853–871.
[4] V. DRUSKIN AND L. KNIZHNERMAN, *Extended Krylov subspaces: Approximation of the matrix square root and related functions*, SIAM J. Matrix Anal. Appl., 19 (1998), pp. 755–771.
[5] V. DRUSKIN AND V. SIMONCINI, *Adaptive rational Krylov subspaces for large-scale dynamical systems*, Systems Control Lett., 60 (2011), pp. 546–560.
[6] D. FASINO, *Rational Krylov matrices and QR-steps on Hermitian diagonal-plus-semiseparable matrices*, Numer. Linear Algebra Appl., 12 (2005), pp. 743–754.
[7] R. W. FREUND, *Krylov-subspace methods for reduced-order modeling in circuit simulation*, J. Comput. Appl. Math., 123 (2000), pp. 395–421.
[8] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, 4th ed., Johns Hopkins University Press, Baltimore, 2013.
[9] S. GUGERCIN, A. C. ANTOULAS, AND C. BEATTIE, *$\mathcal{H}_2$ model reduction for large-scale dynamical systems*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 609–638.

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

[10] S. GÜTTEL, *Rational Krylov approximation of matrix functions: Numerical methods and optimal pole selection*, GAMM-Mitt., 36 (2013), pp. 8–31.
[11] M. HOCHBRUCK AND G. STARKE, *Preconditioned Krylov subspace methods for Lyapunov matrix equations*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 156–171.
[12] C. JAGELS AND L. REICHEL, *The extended Krylov subspace method and orthogonal Laurent polynomials*, Linear Algebra Appl., 431 (2009), pp. 441–458.
[13] ———, *Recursion relations for the extended Krylov subspace method*, Linear Algebra Appl., 434 (2011), pp. 1716–1732.
[14] K. JBILOU AND A. J. RIQUET, *Projection methods for large Lyapunov matrix equations*, Linear Algebra Appl., 415 (2006), pp. 344–358.
[15] L. KNIZHNERMAN AND V. SIMONCINI, *A new investigation of the extended Krylov subspace method for matrix function evaluations*, Numer. Linear Algebra Appl., 17 (2010), pp. 615–638.
[16] ———, *Convergence analysis of the extended Krylov subspace method for the Lyapunov equation*, Numer. Math., 118 (2011), pp. 567–586.
[17] T. MACH, M. S. PRANIĆ, AND R. VANDEBRIL, *Computing approximate extended Krylov subspaces without explicit inversion*, Electron. Trans. Numer. Anal., 40 (2013), pp. 414–435. http://etna.math.kent.edu/vol.40.2013/pp414-435.dir
[18] T. MACH, M. VAN BAREL, AND R. VANDEBRIL, *Inverse eigenvalue problems linked to rational Arnoldi, and rational (non)symmetric Lanczos*, J. Comput. Appl. Math., (2014). In press, DOI: 10.1016/j.cam.2014.03.015.
[19] T. MACH AND R. VANDEBRIL, *On deflations in extended QR algorithms*, SIAM J. Matrix Anal. Appl., 35 (2014), pp. 559–579.
[20] C. C. PAIGE AND M. A. SAUNDERS, *Solution of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal., 12 (1975), pp. 617–629.
[21] A. RUHE, *Rational Krylov sequence methods for eigenvalue computation*, Linear Algebra Appl., 58 (1984), pp. 391–405.
[22] ———, *The Rational Krylov algorithm for nonsymmetric eigenvalue problems, III: Complex shifts for real matrices*, BIT, 34 (1994), pp. 165–176.
[23] ———, *Rational Krylov algorithms for nonsymmetric eigenvalue problems, II: Matrix pairs*, Linear Algebra Appl., 197/198 (1994), pp. 283–296.
[24] ———, *Rational Krylov: A practical algorithm for large sparse nonsymmetric matrix pencils*, SIAM J. Sci. Comput., 19 (1998), pp. 1535–1551.
[25] Y. SAAD, *Krylov subspace methods for solving large unsymmetric linear systems*, Math. Comp., 37 (1981), pp. 105–126.
[26] R. VANDEBRIL, *Chasing bulges or rotations? A metamorphosis of the QR-algorithm*, SIAM J. Matrix Anal. Appl., 32 (2011), pp. 217–247.
[27] R. VANDEBRIL, M. VAN BAREL, AND N. MASTRONARDI, *Matrix Computations and Semiseparable Matrices, Volume I: Linear Systems*, Johns Hopkins University Press, Baltimore, 2008.
[28] R. VANDEBRIL AND D. S. WATKINS, *A generalization of the multishift QR algorithm*, SIAM J. Matrix Anal. Appl., 33 (2012), pp. 759–779.