ON COMPUTING EFFICIENT DATA-SPARSE REPRESENTATIONS OF UNITARY PLUS LOW-RANK MATRICES

R. BEVILACQUA†, G.M. DEL CORSO‡, AND L. GEMIGNANI§

Abstract. Efficient eigenvalue solvers for unitary plus low-rank matrices exploit the structural properties of the Hessenberg reduction performed in the preprocessing phase. Recently some two-stage fast algorithms have been proposed for computing the Hessenberg reduction of a matrix $A = D + UV^H$, where $D$ is a unitary $n \times n$ diagonal matrix and $U, V \in \mathbb{C}^{n \times k}$, which carry out an intermediate transformation into a perturbed block CMV form. However these algorithms generally suffer from some inefficiencies due to the compressed representation of the final Hessenberg matrix as product of $O(nk)$ Givens rotations arranged in a Givens-Vector format. In this paper we modify the bulge-chasing technique applied to the block CMV-like matrix in the second phase of the fast Hessenberg reduction scheme in order to provide a structured factored representation of the final Hessenberg matrix as product of lower and upper unitary Hessenberg matrices possibly perturbed in the first $k$ rows. Such data-sparse representation is well suited for fast eigensolvers and it can immediately be used under the QR method.

Key words. Hessenberg reduction, Rank-structured matrices, QR Method, Bulge chasing, CMV matrix, Complexity.

AMS subject classifications. 65F15

1. Introduction. Let $A = D + UV^H$ where $D$ is a unitary $n \times n$ diagonal matrix and $U, V \in \mathbb{C}^{n \times k}$. Such matrices do arise commonly in the numerical treatment of structured (generalized) eigenvalue problems [1, 2]. In particular any unitary plus low-rank matrix can be reduced in this form by a similarity (unitary) transformation. Recently in [12] some fast $O(n^2k)$ algorithms have been developed for reducing $A$ into a Hessenberg form that amounts to the customary preprocessing step toward eigenvalue computation. Such algorithms are two-phase: in the first phase the matrix $A$ is reduced in a banded form $A_1$ employing a block CMV-like format to represent the unitary part. The second phase amounts to incrementally annihilate the lower subdiagonals of $A_1$ by means of Givens rotations which are accumulate in order to construct a data-sparse compressed representation of the final Hessenberg matrix $A_2$. The representation involves $O(nk)$ data storage consisting of $O(n)$ vectors of length $k$ and $O(nk)$ Givens rotations. This compression is usually known as a Givens–Vector representation [16, 17], and it can also be explicitly resolved to produce a generators-based representation [9, 10]. However, a major weakness of this approach is that both these two compressed formats are not suited to be exploited in the design of fast specialized eigensolvers for unitary plus low rank matrices using $O(n^2k)$ ops only.

In this paper we circumvent this drawback by introducing a different data-sparse compressed representation of the final Hessenberg matrix which is effectively usable in fast eigenvalue schemes. Our derivation is based on three key ingredients or building blocks:

1. A suitable extension of the well known factorization of CMV matrices as product of two block diagonal unitary matrices that are both the direct sum of $2 \times 2$ or $1 \times 1$ unitary blocks (compare with [14] and the references given

∗The research of the last two authors was partially supported by GNCS project “Tecniche innovative per problemi di algebra lineare” and by the project sponsored by University of Pisa under the grant PRA-2017-05.
†Dipartimento di Informatica, Università di Pisa, Pisa, Italy, roberto.bevilacqua@unipi.it
‡Dipartimento di Informatica, Università di Pisa, Pisa, Italy, gianna.delcorso@unipi.it
§Dipartimento di Informatica, Università di Pisa, Pisa, Italy, luca.gemignani@unipi.it

arXiv:1901.08411v1 [math.NA] 24 Jan 2019
therein). Specifically, block CMV matrices with blocks of size $k$ are $2k$-banded unitary matrices allowing a ‘staircase-shaped’ profile. It is shown that a block CMV matrix with blocks of size $k$ admits a factorization as product of two unitary block diagonal matrices with $k \times k$ diagonal blocks. It follows that the block CMV matrix can be decomposed as the product of a unitary lower $k$--Hessenberg matrix multiplied by a unitary upper $k$--Hessenberg matrix.

2. An embedding technique which for a given block CMV matrix plus a rank--$k$ correction located in the first $k$ rows makes possible to construct a larger matrix $\hat{A} \in \mathbb{C}^{(n+k) \times (n+k)}$ which is still unitary plus rank--$k$, block triangular and, moreover, it can be factored as $\hat{A} = L \cdot F \cdot R$, where $L$ is the product of $k$ unitary lower Hessenberg matrices, $R$ is the product of $k$ unitary upper Hessenberg matrices and the middle factor $F$ is unitary plus rank--$k$ with some additional symmetries.

3. A theoretical result which provides conditions under which a matrix specified in the form $\hat{A} = L \cdot F \cdot R$ turns out to be Hessenberg.

Combining together these ingredients allows the design of a specific bulge-chasing strategy for converting the LFR factored representation of $\hat{A}$ into the LFR decomposition of $A$ in such a way that $A$ is upper Hessenberg. The final representation of $\hat{A}$ thus involves $O(nk)$ data storage consisting of $O(k)$ vectors of length $n$ and $O(nk)$ Givens rotations. Furthermore, the representation is eligible as input for the fast eigensolver for unitary plus low rank matrices developed in [5].

This scheme can be applied to block companion matrices as well, just skipping the representation of the unitary part as block CMV matrices. This extends the range of possible applications of the methodology proposed in this paper to a wider class of interesting problems.

The paper is organized as follows. In Section 2 we first describe the block analogue of CMV matrices and its factored LFR representation and, then we review the algorithm from [12] for transforming a unitary diagonal plus rank--$k$ matrix into a block CMV matrix plus a rank--$k$ correction located in the first $k$ rows. In Section 3 we investigate the properties of LFR representations of unitary plus rank--$k$ Hessenberg matrices. In Section 4 we present our algorithm which modifies the LFR representation of a block CMV matrix plus a rank--$k$ correction located in the first $k$ rows by computing the corresponding LFR representation of a unitarily similar Hessenberg matrix. Finally, numerical experiments are discussed in Section 5 whereas conclusions and future work are drawn in Section 6.

2. Block CMV Matrices: Properties and Reductions. A block analogue of the CMV form of unitary matrices has been introduced in [3,12].

Definition 1 (CMV shape). A unitary matrix $A \in \mathbb{C}^{n \times n}$ is said to be CMV structured with block size $k$ if there exist $k \times k$ non-singular matrices $R_i$ and $L_i$, respectively upper and lower triangular, such that

$$
A = \begin{bmatrix}
L_3 & \times & \times \\
R_1 & \times & \times \\
\times & \times & \times \\
R_2 & \times & \times \\
\times & \times & \\
R_4 & \times & \ddots \\
& & \ddots \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
& & \\
where the symbol $\times$ has been used to identify (possibly) nonzero blocks.

In order to simplify the notation we often assume that $n$ is a multiple of $2k$, so the above structures fit “exactly” in the matrix. However, this is not restrictive and the theory presented here continue to hold in greater generality. In practice, one can deal with the more general case by allowing the blocks in the bottom-right corner of the matrix to be smaller. Notice that a matrix in CMV form with blocks of size $k$ is, in particular, $2k$-banded. The CMV structure with blocks of size 1 has been proposed as a generalization of what the tridiagonal structure is for Hermitian matrices in [7] and [13].

A further analogy between the scalar and the block case is derived from the Nullity Theorem [11] that is here applied to unitary matrices.

**Lemma 2 (Nullity Theorem).** Let $U$ be a unitary matrix of size $n$. Then

$$
\text{rank}(U(\alpha, \beta)) = \text{rank}(U(J \setminus \alpha, J \setminus \beta)) + |\alpha| + |\beta| - n
$$

where $J = \{1, 2, \ldots, n\}$ and $\alpha$ and $\beta$ are subsets of $J$. If $\alpha = \{1, \ldots, h\}$ and $\beta = J \setminus \alpha$ we have

$$
\text{rank}(U(1: h, h + 1:n)) = \text{rank}(U(h + 1:n, 1:h)), \quad \text{for all } h = 1, \ldots, n - 1.
$$

From Lemma 2 applied to a block CMV structured matrix $A$ of block size $k$ we find that for for $p > 0$:

$$
0 = \text{rank}(A[1: 2pk, (2p + 1)k + 1:n]) = \text{rank}(A[2pk + 1:n, 1:(2p + 1)k]) - k
$$

which gives

$$
\text{rank}(A[2pk + 1: 2p + 1)k, (2p - 1)k + 1:(2p + 1)k]) = k.
$$

Pictorially we are setting rank constraints on the following blocks

$$
A = \begin{bmatrix}
\times & \times & L_3 \\
R_1 & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\vdots & \vdots & \vdots \\
\end{bmatrix}
$$

and by similar arguments on the corresponding blocks in the upper triangular portion.

In the scalar case with $k = 1$ these conditions make possible to find a factorization of the CMV matrix as product of two block diagonal matrices usually referred to as the classical Schur parametrization [6]. Similarly, here we introduce a block counterpart of the Schur parametrization which gives a useful tool to encompass the structural properties of block CMV representations.

**Lemma 3 (CMV factorization).** Let $A$ be a unitary CMV structured matrix with blocks of size $k$ as defined in Definition 1. Then $A$ can be factored in two block diagonal unitary matrices $A = A_1A_2$ of the form:

$$
A_1 = \text{diag}(A_{1,1}, \ldots, A_{1,s}), \quad A_2 = \text{diag}(J_k, A_{2,2}, \ldots, A_{2,s+1})
$$
such that \( A_{2,j+1} \) have \( k \) rows and columns and all the other blocks \( A_{i,j} \) have \( 2k \) rows and columns and bandwidth \( k \) with both \( A_{i,j}(k+1:2k,1:k) \) and \( A_{i,j}(1:k,k+1:2k) \) triangular matrices of full rank. Moreover, each matrix \( A \) admitting such a factored form is in turn CMV.

**Proof.** The proof of this result is constructive, and can be obtained by performing a block QR decomposition. We notice that if we compute a QR decomposition of the top-left \( 2k \times k \) block of \( A \) we have

\[
\begin{bmatrix}
Q_{1,1} & Q_{1,2} \\
R_{2,1} & Q_{2,2}
\end{bmatrix}
\begin{bmatrix}
I \\
I
\end{bmatrix}
= 
\begin{bmatrix}
\tilde{X} \\
\tilde{X}
\end{bmatrix}
\]

where \( \tilde{X} \) identifies the blocks that have been altered by the transformation and the block in position \((1,1)\) can be assumed to be the identity matrix. Notice that in the first row the blocks in the second and third columns have to be zero due to \( A \) being unitary, and that the \( R_{2,1} \) block is nonsingular upper triangular since it inherits the properties of \( R_1 \).

We can continue this process by computing the QR factorization of \( \begin{bmatrix} \tilde{X} & \times \end{bmatrix} \). Notice that, from the application of the Nullity Theorem 2 the block identified by \( \begin{bmatrix} \tilde{X} & \times \end{bmatrix} \) in the picture has rank at most \( k \). This also holds for all the other blocks for the same kind. In particular, computing the QR factorization of the first \( k \) columns and left-multiplying by \( Q^H \) will put to zero also the block on the right of \( R_2 \). We will then get the following factorization:

\[
\begin{bmatrix}
Q_{1,1} & Q_{1,2} & Q_{1,3} & Q_{1,4} \\
R_{2,1} & Q_{2,2} & Q_{3,3} & Q_{4,4} \\
R_{4,3} & R_{4,4} & R_{4,5} & R_{4,6} \\
\end{bmatrix}
\begin{bmatrix}
I \\
I \\
I \\
I
\end{bmatrix}
= 
\begin{bmatrix}
\tilde{X} \\
\tilde{X} \\
\tilde{X} \\
\tilde{X}
\end{bmatrix}
\]

where we notice that, as before, the block \( R_{4,3} \) is nonsingular upper triangular and that some blocks in the upper part have been set to zero thanks to the unitary property. The process can then be continued until the end of the matrix, providing a factorization of \( A \) as product of two unitary block diagonal matrices, that is \( A = \tilde{A}_1 \tilde{A}_2 \).

This factorization can further be simplified by means of a block diagonal scaling

\( A = (\tilde{A}_1 D)(D^H \tilde{A}_2) = A_1 A_2 \) with \( D = \text{diag}(D_1, \ldots, D_{2n}) \), \( D_{2j-1} = I_k \) and \( D_{2j} \) \( k \times k \) unitary matrices determined so that the blocks \( A_{i,j} \) are of bandwidth \( k \). For the sake of illustration consider \( j = 1 \) and let \( Q_{1,2}^H = QR \) be a QR decomposition of \( Q_{1,2}^H \). By setting \( D_2 = Q \) we obtain that \( Q_{1,2} D_2 = R^H \) and, moreover, from \( L_3 = Q_{1,2} D_2 (A_2)_{2,3} = R^H (A_2)_{2,3} \) it follows that the block of \( A_2 \) in position \((2,3)\) also exhibits a lower triangular structure. The construction of the remaining blocks \( D_{2j}, j > 1 \), proceeds in a similar way.
Pictorially, the above result gives the following structure of $A_1$ and $A_2$:

$$
A_1 = \begin{bmatrix}
\end{bmatrix}, \quad A_2 = \begin{bmatrix}
\end{bmatrix}
$$

The interest toward the properties of block CMV matrices is renewed in [12] where a general scheme is proposed to transform a unitary diagonal plus a rank–$k$ matrix into a block CMV structured matrix plus a rank–$k$ perturbation located in the first $k$ rows only. More specifically we have the following [12].

**Theorem 4.** Let $D \in \mathbb{C}^{n \times n}$ be a unitary diagonal matrix and $U \in \mathbb{C}^{n \times k}$ of full rank $k$ with $n = 2sk$ for some $s \in \mathbb{N}$. Then, there exists a unitary matrix $P$ such that $A = PD^PH$ is CMV structured with block size $k$ and $PU = (e_1 \otimes I_h)U_1$.

By applying Theorem 4 to the matrix pair $(D^H, U)$ we find that there exists a unitary matrix $P$ such that $A = PD^HP^H$ is CMV structured with block size $k$ and $PU = (e_1 \otimes I_h)U_1$. In view of Lemma 3 this yields

$$P(D + UV^H)P^H = A^H + (e_1 \otimes I_h)U_1(PV)^H = \text{diag}(I_k, A_{2,2}^H, \ldots, A_{2,s+1}^H)(I + (e_1 \otimes I_h)Z^H)\text{diag}(A_{1,1}^H, \ldots, A_{1,s}^H).$$

Since the left-hand and the right-hand side matrices are unitary $k$–banded it follows that they can both be factored as the product of $k$ unitary Hessenberg matrices. Summing up for a given matrix pair $(D, U)$, where $D$ is $n \times n$ unitary diagonal and $U \in \mathbb{C}^{n \times k}$ is full rank, one can compute a unitary matrix $P$ such that

$$P(D + UV^H)P^H = L(I + (e_1 \otimes I_h)Z^H)R = LFR$$

where $L$ is the product of $k$ unitary lower Hessenberg matrices, $R$ is the product of $k$ unitary upper Hessenberg matrices and the middle factor $F$ is the identity matrix perturbed in the first $k$ rows. The overall cost of computing this condensed representation is $O(n^2k)$ flops using $O(nk)$ memory storage. In the next sections we investigate the properties of the Hessenberg reduction of a matrix given in the LFR format.

### 3. Factored Representations of Hessenberg Matrices.

In this section we investigate suitable conditions under which a factored representation $A = LFR \in \mathbb{C}^{m \times m}$, where $L$ is the product of $k < n$ unitary lower Hessenberg matrices, $R$ is the product of $k$ unitary upper Hessenberg matrices and the middle factor $F$ is unitary plus rank–$k$ specifies a matrix in Hessenberg form. A key ingredient is the properness of the generalized Hessenberg factors.

**Definition 5.** A matrix $H \in \mathbb{C}^{m \times m}$ is called $k$–upper Hessenberg if $h_{ij} = 0$ when $i > j + k$. Similarly, $H$ is called $k$–lower Hessenberg if $h_{ij} = 0$ when $j > i + k$. In addition, when $H$ is $k$–upper Hessenberg ($k$–lower Hessenberg) and the outermost entries are non-zero, that is, $h_{j+k,j} \neq 0$ ($h_{j+k,j} \neq 0$), $1 \leq j \leq m - k$, then the matrix is called proper.

Note that for $k = 1$ a Hessenberg matrix $H$ is proper iff it is unreduced. Also, a $k$–upper Hessenberg matrix $H \in \mathbb{C}^{m \times m}$ is proper iff $\det(H(k+1 : m, 1 : m-k)) \neq 0$. Similarly a $k$–lower Hessenberg matrix $H$ is proper iff $\det(H(1 : m-k, k+1 : m)) \neq 0$. 


Another basic property of unitary plus rank\( -k\) matrices is the existence of suitable embeddings which maintain their structural properties. The embedding turns out to be crucial to ensure the properness of the factor \( L \) and guarantee the safe application of implicit \( QR \) iterations. The following result is first proved in [5] and here specialized to a matrix of the form given in (1).

**Theorem 6.** Let \( A \in \mathbb{C}^{n \times n} \) be such that \( A = L(I + (e_1 \otimes I_k)Z^H)R = LFR \), where \( L \) and \( R \) are unitary and \( Z \in \mathbb{C}^{n \times k} \). Let \( Z = QG, G \in \mathbb{C}^{k \times k} \), be the economic \( QR \) factorization of \( Z \). Let \( \hat{U} \in \mathbb{C}^{m \times m}, m = n + k \), be defined as

\[
\hat{U} = I_m - \begin{bmatrix} Q & -I_k \\ -I_k & Q \end{bmatrix}^H.
\]

Then it holds

1. \( \hat{U} \) is unitary;
2. the matrix \( \hat{A} \in \mathbb{C}^{m \times m} \) given by

\[
\hat{A} = \begin{bmatrix} L & I_k \\ I_k & \end{bmatrix} \left( \hat{U} + \begin{bmatrix} G^H & \end{bmatrix} + \begin{bmatrix} Q & -I_k \\ -I_k & Q \end{bmatrix}^H \right) \begin{bmatrix} 0 & I_k \\ I_k & 0 \end{bmatrix} \begin{bmatrix} Q & 0 \\ 0 & Z \end{bmatrix}^H.
\]

Proof. Property 1 follows by direct calculations from

\[
\begin{bmatrix} Q & -I_k \\ -I_k & Q \end{bmatrix}^H = 2I_k.
\]

For Property 2 we find that

\[
\hat{U} + \left( \begin{bmatrix} G^H & \end{bmatrix} + \begin{bmatrix} Q & -I_k \\ -I_k & Q \end{bmatrix}^H \right) \begin{bmatrix} I_n & Q \\ 0 & 0_k \end{bmatrix} = \begin{bmatrix} I_n & Q \\ 0 & 0_k \end{bmatrix} + \begin{bmatrix} I_k & 0 \\ 0 & 0 \end{bmatrix} \end{bmatrix}^H.
\]

The unitary matrices \( L \) and \( R \) given in (1) are \( k \)-Hessenberg matrices. The same clearly holds for the larger matrices \( \text{diag}(L, I_k) \) and \( \text{diag}(R, I_k) \) occurring in the factorization of \( \hat{A} \). The next result is the main contribution of this section and it provides conditions under which a matrix specified in the form \( LFR \), where \( L \) is a proper unitary \( k \)-lower Hessenberg matrix and \( F \) is a unitary matrix plus a rank\( -k \) correction, is in Hessenberg form.

**Theorem 7.** Let \( L, R \in \mathbb{C}^{m \times m}, m = n + k \), be two unitary matrices, where \( L \) is a proper unitary \( k \)-lower Hessenberg matrix and \( R \) is a unitary \( k \)-upper Hessenberg matrix. Let \( Q \) be a block diagonal unitary upper Hessenberg matrix of the form \( Q = \begin{bmatrix} I_k & \end{bmatrix} \begin{bmatrix} Q \end{bmatrix} \), with \( Q \) \( n \times n \) unitary Hessenberg. Let \( F = Q + TZ^H \) be a unitary plus rank\( -k \) matrix with \( T, Z \in \mathbb{C}^{m \times k}, T = [T_k, 0]^T \) and \( T_k \) upper triangular and invertible. Suppose that the matrix \( \hat{A} = LFR \) satisfies the block structure

\[
\hat{A} = \begin{bmatrix} A & * \\ 0_{k,n} & 0_{k,k} \end{bmatrix}.
\]

Then \( \hat{A} \) is an upper Hessenberg matrix.
Proof. From Lemma 2 we find that \( M = L(n + 1 : m, 1 : k) \) is nonsingular due to the properness of \( L \).

Now, let us consider the matrix \( C = LQ \). This matrix is unitary with a \( k \)-quasiseparable structure below the \( k \)-th upper diagonal. Indeed, for any \( h, h = 2, \ldots n + 1 \) we have

\[
C(h : m, 1 : h + k - 2) = L(h : m, :) Q(:, 1 : h + k - 2) = L(h : m, 1 : h + k - 1) Q(1 : h + k - 1, 1 : h + k - 2).
\]

Applying Lemma 2 we have \( \text{rank}(L(h : m, 1 : h + k - 1)) = k \), implying that also \( \text{rank}(C(h : m, 1 : h + k - 2)) \leq k \). Since \( C(n + 1 : m, 1 : k) = L(n + 1 : m, :) Q(:, 1 : k) = M \) is non singular, we conclude that \( \text{rank}(C(h : m, 1 : h + k - 2)) = k \), \( 2 \leq h \leq n + 1 \).

From this observation we can then find a set of generators \( P, S \in \mathbb{C}^{(m \times k)} \) and a \((1 - k)\)-upper Hessenberg matrix \( U_k \) such that \( U_k(1, k) = U_k(n, m) = 0 \) so that \( C = P S^H + U_k \) [8].

Then we can recover the rank \( k \) correction \( P S^H \) from the left-lower corner of \( C \) obtaining

\[
P S^H = C(:, 1 : k) M^{-1} C(n + 1 : m, :) = L(:, 1 : k) M^{-1} C(n + 1 : m, :),
\]

since \( C(:, 1 : k) = L Q(:, 1 : k) = L(:, 1 : k) \). Notice that \( B = U_k R \) is upper Hessenberg as it is the product of a \((1 - k)\)-upper Hessenberg matrix by a \( k \)-upper Hessenberg matrix. Moreover, we find that \( B(n + 1 : m, :) = U_k(n + 1 : m, :) R = 0 \) since \( U_k(n + 1 : m, :) = 0 \). From the block structure of \( \hat{A} \) there follows that

\[
(C(n + 1 : m, :) + M T_k Z^H) R = 0,
\]

which gives

\[
P S^H = L(:, 1 : k) M^{-1} C(n + 1 : m, :) = -L(:, 1 : k) T_k Z^H = -L T Z^H.
\]

Hence \( U_k = L(Q + T Z^H) = L F \) and therefore \( B = U_k R = L F R = \hat{A} \) which concludes the proof.

\[\square\]

4. The Bulge Chasing Algorithm. In this section we present a bulge-chasing algorithm to compute the Hessenberg reduction of the matrix \( \hat{A} \) given as in Theorem 6. In Section 2 we described as any unitary diagonal plus low rank matrix can be brought to the form

\[
L(I + (e_1 \otimes I_k) Z^H) R,
\]

but there are other matrices that naturally can be represented in this factored form such as the block companion matrices. Block companion matrices are of the form \( U + (c_1 \otimes I_k) P^H \) where \( U = C \otimes I_k \) and \( C \) is the down-shift matrix. For this important class of matrices we can avoid the preprocessing steps and obtain immediately the representation in (2) choosing \( L = I, R = U \) and \( Z = U P \).

Let us set

\[
X = \begin{bmatrix} Q \\ -I_k \end{bmatrix}, \quad Y = \begin{bmatrix} G^H \\ 0 \end{bmatrix} + X, \quad W = \begin{bmatrix} Q \\ 0 \end{bmatrix},
\]

so that we have

\[
\hat{A} = \begin{bmatrix} L \\ I_k \end{bmatrix} (\hat{U} + Y W^H) \begin{bmatrix} R \\ I_k \end{bmatrix}, \quad \hat{U} = I_m - X X^H.
\]
Observe that $X(k + 1 : m, :) = Y(k + 1 : m, :)$ and, moreover, $Y(n + 1 : m, :) = -I_k$ which implies $\text{rank}(Y) = k$. In the preprocessing phase we initialize

$$L_0 = \begin{bmatrix} L & I_k \\ I_k & 0 \end{bmatrix}, \quad R_0 = \begin{bmatrix} R & I_k \\ I_k & 0 \end{bmatrix}, \quad X_0 = X, \quad Y_0 = Y, \quad W_0 = W.$$

Notice that $L_0$ is a unitary $k$-lower Hessenberg matrix and $R_0$ is a unitary $k$-upper Hessenberg matrix and, therefore, they can both be represented by the product of $k$ Hessenberg matrices. This property will be maintained under the bulge chasing process.

The reduction of $\hat{A} = \hat{A}_0$ in Hessenberg form proceeds in three steps according to Theorem 7. The first two steps amounts to determine a different representation of the same matrix $\hat{A}_0$. The third step is a bulge-chasing scheme to complete the Hessenberg reduction.

1. *QR decomposition of $Y_0$* We compute the full QR factorization of $Y_0 = Q_0 T_0$. Since $Y_0$ is full rank the matrix $\hat{T}_0 = T_0 (1 : k, :)$ is invertible and, moreover, the matrix $Q_0$ can be takes ad a $k$-lower Hessenberg proper matrix (see Lemma 2.4 of [5]). We can write

$$\hat{A}_0 = (L_0 Q_0) \cdot (Q_0^H \hat{U} + T_0 W_0^H) \cdot R_0.$$

Then the matrix $\hat{A}_1 := L_0^H \hat{A}_0 L_0$ is such that

$$\hat{A}_1 = Q_0 \cdot (Q_0^H \hat{U} R_0 + T_0 W_0^H R_0) L_0.$$

Notice that $\hat{U}_1 = Q_0^H \hat{U} R_0$ is a unitary $2k$-upper Hessenberg matrix. Indeed, we have that $\hat{U}_1 = Q_0^H \hat{U} Q_0 Q_0^H R_0 = (I_m - \hat{X} \hat{X}^H) Q_0^H R_0$, where $\hat{X} = Q_0^H X$ and $\hat{X}(2k + 1 : m, :) = -Q_0^H (2k + 1 : m, 1 : k) G^H = 0$. Therefore, it holds $\hat{U}_1 = ((I_{2k} - \hat{X}(1 : 2k, :) \hat{X}^H(:, 1 : 2k)) \oplus I_{m - 2k}) Q_0^H R_0$ which, for the block diagonal structure of $I_m - \hat{X} \hat{X}^H$, turns out to be $2k$-upper Hessenberg.

2. *Block decomposition of $\hat{U}_1$* We compute the full QR factorization of $\hat{U}_1^H(:, 1 : k)$. Specifically we determine a unitary matrix $P$ such that $\hat{U}_1 (1 : k, :) P = [I_k, 0]$, and such $P$ can be taken in $k$-lower Hessenberg form (see Lemma 2.4 of [5]). The matrix

$$\hat{U}_1 P = \begin{bmatrix} I_k \\ U_1 (k + 1 : m, :) P(:, k + 1 : m) \end{bmatrix} = \begin{bmatrix} I_k \\ Q \end{bmatrix} \quad (4)$$

where $Q$ is a unitary $k$-upper Hessenberg matrix, due to the fact that $U_1 (k + 1 : m, :)$ is $k$-upper Hessenberg and $P(:, k + 1 : m)$ is lower triangular. We obtain that

$$\hat{A}_1 = Q_0 \cdot (\hat{U}_1 P + T_0 W_0^H R_0 P) P^H L_0,$$

which gives

$$\hat{A}_2 = L_0 \hat{A}_1 L_0^H = \hat{A}_0 = (L_0 Q_0) \cdot (\hat{U}_1 P + T_0 W_0^H R_0 P) P^H.$$

Since $L_0$ is $k$-banded we can factorize $L_0 Q_0 = Q_1 L_1$ where $Q_1$ is a unitary $k$-lower Hessenberg matrix and $L_1 = \begin{bmatrix} I_k \\ L_1 \end{bmatrix}$ where $L_1$ is a unitary $k$-upper Hessenberg matrix. The details of such factorization are reported in Section 5. It follows that

$$\hat{A}_0 = Q_1 \cdot (L_1 \hat{U}_1 P + T_0 W_0^H R_0 P) P^H. \quad (5)$$
The matrix $\tilde{U}_2 = L_1 \tilde{U}_1 P$ satisfies $\tilde{U}_2 = \begin{bmatrix} I_k & \ast \\ 0 & \tilde{U}_2 \end{bmatrix}$ where $\tilde{U}_2$ is a unitary 2$k$-upper Hessenberg matrix. Observe that $Q_0(n + 1 : m, 1 : k) = Q_1(n + 1 : m, 1 : k)$ and, moreover $Q_0(n + 1 : m, 1 : k)$ is nonsingular, because $Q_0$ is proper. From Lemma 2 this implies the properness of $Q_1$. This property is maintained in the subsequent steps of the reduction process so that the final matrix is guaranteed to be proper as prescribed in Theorem 7.

3. (Hessenberg reduction of $\tilde{U}_2$) The third phase of the Hessenberg reduction consists of reducing the inner matrix $\tilde{U}_2$ in Hessenberg form by means of a bulge-chasing procedure. For the sake of illustration let us consider the first step. Let us determine a unitary upper Hessenberg matrix $\mathcal{G}_1 \in \mathbb{C}^{2k \times 2k}$ such that

$$\mathcal{G}_1 \tilde{U}_2(2 : 2k + 1, 1) = \alpha_1 e_1.$$ 

The application of $\mathcal{G}_1^H$ on the right of the matrix $\mathcal{Q}_1$ by computing $Q_1(:, k + 2 : 3k + 1)\mathcal{G}_1^H$ creates a bulge formed by an additional segment above the last nonzero superdiagonal of $\mathcal{Q}_1$. This segment can be annihilated by a novel unitary upper Hessenberg matrix $\mathcal{G}_2 \in \mathbb{C}^{2k \times 2k}$ working on the left of

$$Q_1(:, k + 2 : 3k + 1)\mathcal{G}_1^H$$

by acting on the rows of index $2$ through index $2k + 1$. The application of $\mathcal{G}_2^H$ on the right of $\mathcal{P}^H$ produces a bulge which can be zeroed by a unitary upper Hessenberg matrix $\mathcal{G}_3 \in \mathbb{C}^{2k \times 2k}$ working on rows from $k + 2$ to $3k + 1$. Finally, the matrix

$$\tilde{U}_2 \leftarrow (1 \oplus \mathcal{G}_1 \oplus I_{n-2k-1})\tilde{U}_2(1 \oplus \mathcal{G}_3^H \oplus I_{n-2k-1})$$

has a bulge on the rows of index $2k + 2$ through index $4k + 1$ which can be chased away by a sequence of $O(n/k)$ transformations having the same structure as above.

The cost analysis is rather standard for matrix algorithms based on chasing operations [4].

1. Step 1 requires to compute the economic QR decomposition of a matrix of size $(n + k) \times k$ and to multiply a unitary $k$-Hessenberg matrix specified as product of $k$ unitary Hessenberg matrices by $k$ vectors of size $n + k$. The total cost is $O(nk^2)$ ops.

2. The cost of Step 2 is asymptotically the same. The construction of the factored representation of $\hat{Q}$ as well as the computation of $L_1$ and $Q_1$ can still be performed using $O(nk^2)$ ops.

3. The dominant cost is the execution of Step 3. The zeroing of the sub-diagonal entries costs $O(n^2k^2) = O(n^2k)$ ops.

In the next section we discuss the results of numerical experiments confirming the effectiveness and the robustness of our proposed approach.

5. Numerical Results. The structured Hessenberg reduction scheme described in the previous section has been implemented using MatLab for numerical testing. The resulting algorithm basically amounts to manipulate chains of unitary upper Hessenberg matrices. Each unitary upper Hessenberg matrix $H \in \mathbb{C}^{m \times m}$ is represented as product of elementary transformations, i.e., $H = \mathcal{G}_1 \mathcal{G}_2 \cdots \mathcal{G}_{m-1} \mathcal{D}_m$ where $\mathcal{G}_\ell = I_{\ell-1} \oplus G_\ell \oplus I_{m-\ell-1}$ with $G_\ell = \begin{bmatrix} \alpha_\ell & \beta_\ell \\ -\bar{\beta}_\ell & \bar{\alpha}_\ell \end{bmatrix}$, $|\alpha_\ell|^2 + |\beta_\ell|^2 = 1$, $\alpha_\ell, \beta_\ell \in \mathbb{C}$, are unitary Givens rotations and $\mathcal{D}_m = I_{m-1} \oplus \theta_m$ with $|\theta_m| = 1$. In this way the matrix $H$ is stored by two vectors of length $m$ formed by the elements $\alpha_\ell, \beta_\ell$, $1 \leq \ell \leq m - 1$. The cost analysis is rather standard for matrix algorithms based on chasing operations [4].
and \( \theta_m \). The same representation also extends to unitary \( k \)-upper Hessenberg matrices specified as the product of \( k \) unitary upper Hessenberg matrices multiplied on the right by a unitary diagonal matrix which is the identity matrix modified in the last diagonal entry.

At step 1 of the structured Hessenberg reduction scheme we first compute the full QR factorization of the matrix \( Y_0 \in \mathbb{C}^{n \times k} \). The matrix \( Q_0^{\ell} \) turns out to be the product of \( k \) unitary upper Hessenberg matrices. Then we have to incorporate the unitary matrix \( S = I_{2k} \times (1 \div 2k,:)X^H(:,1 \div 2k) \) on the right into the factored representations of \( Q_0^{\ell} \) and \( R_0 \). To do this we can decompose \( S \) as product of at most \( k(2k-1) \) elementary unitary transformations of size \( 2 \times 2 \) and add each single transformation one by one on the right to the factored representations of \( Q_0^{\ell} \) and \( R_0 \) by a sequence of turnover and fusion operations on the sequences of elementary transformations in \( Q_0^{\ell} \) and \( R_0 \) (see [15] for the detailed description of these operations on elementary transformations).

At the beginning of step 2 the matrix \( \hat{U}_1 \) is determined by the product of two unitary \( k \)-upper Hessenberg matrices, say \( \hat{U}_1 = \hat{P}\hat{Q} \). To reshape this factorization in the desired form in equation (4) we have to move each elementary transformation of \( \hat{Q} \) on the left. By applying the elementary transformation on the right to the matrix \( \hat{P} \) we create a bulge which can be annihilated by an elementary transformation on the left of the form \( G_{\ell} \) with \( \ell > k \). In this way we find \( \hat{P}\hat{Q} = \hat{Q}\tilde{P} \) where \( \tilde{Q} = \begin{bmatrix} I_k & \hat{Q} \end{bmatrix} \) is the matrix appearing in (4). Since \( \hat{Q} \) is formed by \( O(nk) \) elementary transformations the reshaping costs \( O(nk^2) \) ops. With a similar reasoning we can compute the representations of \( Q_1 \) and \( L_1 \).

The third phase of the structured Hessenberg reduction scheme basically amounts to reduce the matrix \( \hat{U}_2 = L_1\tilde{Q} \) into a matrix of the form \( \begin{bmatrix} I_k & \hat{U}_2 \end{bmatrix} \), with \( \hat{U}_2 \) \( n \times n \) unitary Hessenberg. To be specific assume that \( L_1 = L_{1,1} \cdots L_{1,k} \) and \( \tilde{Q} = \tilde{Q}_1 \cdots \tilde{Q}_k \), where \( L_{1,j} \) and \( \tilde{Q}_j \) are unitary upper Hessenberg matrices with the leading principal submatrix of order \( k \) equal to the identity matrix. The overall reduction process splits into \( n \) intermediate steps. At each step the first active elementary transformations of \( \tilde{Q}_k, \ldots, \tilde{Q}_1, \tilde{L}_1,k, \ldots, \tilde{L}_1,1 \) are annihilated. Each transformation is moved on the left by creating a bulge in the leftmost factor \( \tilde{Q}_1 \). This bulge is removed by applying a similarity transformation.

Let us consider the first step. Let \( L_{1,i} = G_{k+1}^{(i)} \cdots G_{m-1}^{(i)}D_m^{(i)} \) denote the Schur parametrization of \( L_{1,i} \) and similarly let \( \tilde{Q}_i = H_{k+1}^{(i)} \cdots H_{m-1}^{(i)}E_m^{(i)} \) that of \( \tilde{Q}_i \). At this step we move left the first elementary transformation of each factor of the product \( L_1\tilde{Q} \) as follows\(^1\): \n
\[
L_1\tilde{Q} = (G_{k+1}^{(1)} \cdots G_{m-1}^{(1)}) \cdots (G_{k+1}^{(k)} \cdots G_{m-1}^{(k)})(H_{k+1}^{(1)} \cdots H_{m-1}^{(1)}) \cdots (H_{k+1}^{(k)} \cdots H_{m-1}^{(k)})D =
\begin{bmatrix}
\tilde{H}_{k+1,1}^{(k)} & \cdots & \tilde{H}_{k+1,m}^{(k)} & \cdots & \tilde{H}_{m-1,1}^{(k)} & \cdots & \tilde{H}_{m-1,m}^{(k)}
\end{bmatrix}
\begin{bmatrix}
\tilde{G}_{k+1,1}^{(k)} & \cdots & \tilde{G}_{k+1,m}^{(k)} & \cdots & \tilde{G}_{m-1,1}^{(k)} & \cdots & \tilde{G}_{m-1,m}^{(k)}
\end{bmatrix}
\begin{bmatrix}
\tilde{L}_{1,1} & \cdots & \tilde{L}_{1,k} & \cdots & \tilde{L}_{1,m-1}
\end{bmatrix}
\begin{bmatrix}
\tilde{Q}_1 & \cdots & \tilde{Q}_k
\end{bmatrix}
\]

where

\[
\tilde{L}_{1,j} = \tilde{G}_{k+1}^{(j)} \cdots \tilde{G}_{m-1}^{(j)} \quad \text{and} \quad \tilde{Q}_j = \tilde{H}_{k+1}^{(j)} \cdots \tilde{H}_{m-1}^{(j)}.
\]

\(^1\)As observed, we can use only a unitary diagonal matrix to keep track of all the diagonal contributions.
At this point we bring the bulge $B$ on the left of $Q_1$ in equation (5) obtaining

$$\hat{A}_0 = \hat{B}Q_1(\hat{U}_2 + T_0W_0^HR_0P)P^H,$$

where $\hat{B} = \Gamma_{2k} \cdot \cdots \cdot \Gamma_2$ is the product of a sequence of elementary transformations in ascending order acting on rows $2 : 2k$. The bulge $\hat{B}$ is removed by chasing an elementary transformation at a time. For example to remove $\Gamma_{2k}$ we apply the similarity transformation $\Gamma_{2k}^H\hat{B}Q_1(+T_0W_0^HR_0P)P^H\Gamma_{2k}$ that will shift down the bulge of $2k$ positions. So $O(n/k)$ chasing step will be necessary to get rid of that first transformation. In this way the overall process is completed using $O(nk \cdot k \cdot n/k) = O(n^2k)$ ops.

Note that the whole similarity transformation acts only on the first $n$ rows leaving untouched the null rows at the bottom of $\hat{A}$ in equation (3).

Numerical experiments have been performed to confirm the computational properties of the proposed method. The CMV reduction of the input unitary diagonal plus rank $−k$ matrix $D + UV^H$ is computed using the algorithm presented in [12] which is fast and backward stable. Our tests focus on the numerical performance of the Hessenberg reduction scheme provided in the previous section given the factors $L, R$ and $Z$ satisfying (1). In the next tables we show the backward errors $\epsilon_P, \epsilon_B$ and $\epsilon_H$ generated by our procedure. These errors are defined as follows:

1. $\epsilon_P$ is the error computed at the end of the first two preparatory steps. Given the matrix $A$ of size $n$ represented as in Theorem 6 we find the matrix $\hat{A}$ of size $m = n + k$ obtained at the end of step 2. Denoting by $fl(\hat{A})$ the computed matrix, the error is

$$\epsilon_P = \frac{\|A - fl(\hat{A}(1: n, 1: n))\|_2}{mk\|A\|_2}.$$

2. $\epsilon_B$ is the classical backward error generated in the final step given by

$$\epsilon_B = \frac{\|H - Qfl(\hat{A})Q^H\|_2}{mk\|A\|_2},$$

where $H$ is the matrix computed by multiplying all the factors obtained at the end of the third step, assuming whereas $Q$ is the product of the unitary transformations acting by similarity on the left and on the right of the matrix $fl(\hat{A})$ in the Hessenberg reduction phase.

3. $\epsilon_H$ is used to measure the Hessenberg structure of the matrix $H$. It is

$$\epsilon_H = \frac{\|\text{tril}(H, -2)\|_2}{mk\|A\|_2},$$

where $\text{tril}(X, K)$ is the matrix formed by the elements on and below the $K$-th diagonal of $X$.

Next tables report these errors for different values of $n, k$ and $\|A\|_2$.

The results of Table 1, 2, 3 and 4 show that the proposed algorithm is numerically backward stable.

In order to confirm the cost analysis of the algorithm we have also performed experiments taking fixed the size of the matrix. For matrices of size 512 with $k$ varying from 2 to 16 we obtain that the measures of elapsed time $t_k$ satisfy

$$\frac{t_4}{t_2} = 2.34, \quad \frac{t_8}{t_4} = 2.16, \quad \frac{t_{16}}{t_8} = 2.08$$

by illustrating the linear growth of the cost with respect to the size of the perturbation.
6. Conclusions and Future Work. In this paper we have presented a novel algorithm for the reduction in Hessenberg form of a unitary diagonal plus rank $-k$ matrix. By exploiting the rank structure of the input matrix this algorithm achieves computational efficiency both with respect to the size of the matrix and the size of the perturbation as well as numerical accuracy. The algorithm complemented with the structured QR iteration described in [5] yields a fast and accurate eigensolver for unitary plus low rank matrices.

REFERENCES

[1] A. Amiraslani, R. M. Corless, and P. Lancaster, Linearization of matrix polynomials expressed in polynomial bases, IMA J. Numer. Anal., 29 (2009), pp. 141–157, https://doi.org/10.1093/imanum/drm051.

[2] P. Arbenz and G. H. Golub, On the spectral decomposition of Hermitian matrices modified by low rank perturbations with applications, SIAM J. Matrix Anal. Appl., 9 (1988), pp. 40–58, https://doi.org/10.1137/0609004.

[3] Y. Arlinski˘ı, Conservative discrete time-invariant systems and block operator CMV matrices, Methods Funct. Anal. Topology, 15 (2009), pp. 201–236.

[4] J. Aurentz, T. Mach, L. Robol, R. Vandebril, and D. S. Watkins, Core-chasing algorithms for the eigenvalue problem, Fundamentals of Algorithms, SIAM, 2018.

[5] R. Bevilacqua, G. M. Del Corso, and L. Gemignani, Fast QR iterations for unitary plus low rank matrices, tech. report, ArXiv:1810.02708v1, 2018.

[6] A. Bunse-Gerstner and L. Elsner, Schur parameter pencils for the solution of the unitary eigenproblem, Linear Algebra Appl., 154/156 (1991), pp. 741–778, https://doi.org/10.1016/0024-3795(91)90402-I.

[7] M. J. Cantero, L. Moral, and L. Velázquez, Five-diagonal matrices and zeros of orthogonal polynomials on the unit circle, Linear Algebra Appl., 362 (2003), pp. 29–56, https://doi.org/10.1016/S0024-3795(02)00457-3.

[8] S. Delvaux and M. Van Barel, Unitary rank structured matrices, J. Comput. Appl. Math., 215 (2008), pp. 49–78, https://doi.org/10.1016/j.cam.2007.03.020.

[9] Y. Eidelman, I. Gohberg, and I. Haimovici, Separable type representations of matrices and fast algorithms. Vol. 1, vol. 234 of Operator Theory: Advances and Applications, Birkhäuser/Springer, Basel, 2014. Basics. Completion problems. Multiplication and inversion algorithms.
Efficient Data-Sparse Representations of Unitary plus Low-Rank Matrices

\[
\begin{array}{cccc}
\text{n} & \| A \|_2 & \epsilon_P & \epsilon_B & \epsilon_H \\
64 & 1.5e+02 & 7.4e-18 & 4.4e-17 & 2.3e-18 \\
128 & 2.9e+02 & 3.2e-18 & 5.6e-17 & 1.2e-18 \\
256 & 5.5e+02 & 2.5e-18 & 9.6e-17 & 4.1e-19 \\
512 & 1.1e+03 & 1.8e-18 & 1.6e-16 & 5.0e-19 \\
\end{array}
\]

Table 3

Backward errors for random matrices with \( k = 4 \)

\[
\begin{array}{cccc}
\text{n} & \| A \|_2 & \epsilon_P & \epsilon_B & \epsilon_H \\
64 & 6.2e+05 & 6.6e-18 & 5.2e-17 & 5.3e-18 \\
128 & 4.9e+06 & 4.5e-18 & 6.8e-17 & 1.8e-18 \\
256 & 3.8e+07 & 2.2e-18 & 9.2e-17 & 5.5e-19 \\
512 & 2.9e+08 & 2.3e-18 & 1.6e-16 & 8.0e-19 \\
\end{array}
\]

Table 4

Backward errors for random matrices of large norm with \( k = 4 \)

[10] Y. Eidelman, I. Gohberg, and I. Haimovici, Separable type representations of matrices and fast algorithms. Vol. 2, vol. 235 of Operator Theory: Advances and Applications, Birkhäuser/Springer Basel AG, Basel, 2014. Eigenvalue method.
[11] M. Fiedler and T. L. Markham, Completing a matrix when certain entries of its inverse are specified, Linear Algebra Appl., 74 (1986), pp. 225-237, https://doi.org/10.1016/0024-3795(86)90125-4, http://dx.doi.org/10.1016/0024-3795(86)90125-4.
[12] L. Gemignani and L. Robol, Fast Hessenberg reduction of some rank structured matrices, SIAM J. Matrix Anal. Appl., 38 (2017), pp. 574–598, https://doi.org/10.1137/16M1107851, https://doi.org/10.1137/16M1107851.
[13] R. Killip and I. Nenciu, CMV: the unitary analogue of Jacobi matrices, Comm. Pure Appl. Math., 60 (2007), pp. 1148–1188, https://doi.org/10.1002/cpa.20160, http://dx.doi.org/10.1002/cpa.20160.
[14] B. Simon, CMV matrices: five years after, J. Comput. Appl. Math., 208 (2007), pp. 120–154, https://doi.org/10.1016/j.cam.2006.10.033, http://dx.doi.org/10.1016/j.cam.2006.10.033.
[15] R. Vandebril, Chasing bulges or rotations? A metamorphosis of the QR-algorithm, SIAM J. Matrix Anal. Appl., 32 (2011), pp. 217–247, https://doi.org/10.1137/100809167, http://dx.doi.org/10.1137/100809167.
[16] R. Vandebril, M. Van Barel, and N. Mastronardi, Matrix computations and semiseparable matrices. Vol. I, Johns Hopkins University Press, Baltimore, MD, 2008. Linear systems.
[17] R. Vandebril, M. Van Barel, and N. Mastronardi, Matrix computations and semiseparable matrices. Vol. II, Johns Hopkins University Press, Baltimore, MD, 2008. Eigenvalue and singular value methods.