EFFICIENT REDUCTION OF COMPRESSED UNITARY PLUS LOW RANK MATRICES TO HESSENBERG FORM*

ROBERTO BEVILACQUA†, GIANNA M. DEL CORSO†, AND LUCA GEMIGNANI†

Abstract. We present fast numerical methods for computing the Hessenberg reduction of a unitary plus low rank matrix $A = G + UV^H$, where $G \in \mathbb{C}^{n \times n}$ is a unitary matrix represented in some compressed format using $O(nk)$ parameters and U and V are $n \times k$ matrices with $k < n$. At the core of these methods is a certain structured decomposition, referred to as an LFR decomposition, of A as a product of three possibly perturbed unitary k Hessenberg matrices of size n. It is shown that in most interesting cases an initial LFR decomposition of A can be computed very cheaply. Then we prove structural properties of LFR decompositions by giving conditions under which the LFR decomposition of A implies its Hessenberg shape. Finally, we describe a bulge chasing scheme for converting the initial LFR decomposition of A into the LFR decomposition of a Hessenberg matrix by means of unitary transformations. The reduction can be performed at the overall computational cost of $O(n^2 k)$ arithmetic operations using $O(nk)$ storage. The computed LFR decomposition of the Hessenberg reduction of A can be processed by the fast QR algorithm presented in [R. Bevilacqua, G. M. Del Corso, and L. Gemignani, *Numer. Math.*, 144 (2020), pp. 23–53] in order to compute the eigenvalues of A within the same costs.

Key words. Hessenberg reduction, rank-structured matrices, QR method, bulge chasing, CMV matrix, complexity

AMS subject classification. 65F15

DOI. 10.1137/19M1280363

1. Introduction. Eigenvalue computations for small rank modifications of unitary matrices represented in some compressed format is a classical topic in structured numerical linear algebra. Matrices of the form $A = D + UV^H$, where D is a unitary $n \times n$ block diagonal matrix and $U, V \in \mathbb{C}^{n \times k}$, $k < n$, 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 and additionally matrices of this form can be directly generated by linearization techniques based on interpolation schemes applied for the solution of nonlinear eigenvalue problems [6, 7, 8, 10, 19]. The class of unitary block upper Hessenberg matrices perturbed in the first block row or in the last block column includes block companion linearizations of matrix polynomials. These matrices are also related with computational problems involving orthogonal matrix polynomials on the unit circle [22, 23]. Constructing the sequence of orthogonal polynomials w.r.t. a different basis modifies the compressed format of the unitary part by replacing the block Hessenberg shape with the block CMV shape [12, 20, 21]. Semi-infinite block upper Hessenberg and CMV unitary matrices are commonly used to represent unitary operators on a separable Hilbert space [3, 13]. Finite truncations of these matrices are unitary block Hessenberg/CMV matrices modified in the last row or column.

In most numerical methods Hessenberg reduction by unitary similarity transformations is the first step towards eigenvalue computation. Recently a fast reduction

†Dipartimento di Informatica, Università di Pisa, Pisa, 56127, Italy (roberto.bevilacqua@unipi.it, gianna.delcorso@unipi.it, luca.gemignani@unipi.it).

algorithm specifically tailored for block companion matrices has been presented in [5] whereas some efficient algorithms for dealing with block unitary diagonal plus small rank matrices have been developed in [18]. In particular, these latter 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 annihilating the lower subdiagonals of A_1 by means of Givens rotations which are gathered 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 [25], and it can also be explicitly resolved to produce a generators-based representation [16]. However, a major weakness of this approach is that both these two compressed formats, to the best of our knowledge, are not directly suited to be exploited in the design of fast specialized eigensolvers for unitary plus low rank matrices using $O(n^2k)$ flops only.

In this paper we describe a novel $O(n^2k)$ backward stable algorithm for computing the Hessenberg reduction of general matrices $A \in \mathbb{C}^{n \times n}$ of the form $A = G + UV^H$, where G is unitary block diagonal or unitary block upper Hessenberg or block CMV with block size $k < n$ and, in the case G is unitary block upper Hessenberg or block CMV, we have the additional requirement that $U = [I_k, 0, \ldots, 0]^T$. These families include most of the important cases arising in applications such as the Frobenius companion linearizations (with G unitary (block) Hessenberg), the linearization of matrix polynomials in the Lagrange basis [1] (with G unitary (block) diagonal). The (block) CMV case arises when dealing with orthogonal polynomials [20, 22, 23].

This algorithm circumvents the drawback of the method proposed in [18] by introducing a different data-sparse compressed representation of the final Hessenberg matrix which is effectively usable in fast eigenvalue schemes. In particular, the representation is suited for the fast eigensolver for unitary plus low rank matrices developed in [9]. Our derivation is based on three key ingredients or building blocks:

1. A condensed representation of the matrix A (or of a matrix unitary similar to A) which can be specified as $A = L(I + (e_1 \otimes I_k)Z^H)R = LFR$, 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.

    In the case matrix G is block upper Hessenberg or block diagonal, we can obtain the LFR representation in a simple way that we clarify in subsections 2.2 and 2.3. In the case G is a unitary block CMV matrix we provide a suitable extension of the well-known factorization of CMV matrices as the product of two block diagonal unitary matrices that are both the direct sum of 2×2 or 1×1 unitary blocks (compare with [21] and the references given 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 a product of two unitary block diagonal matrices with $k \times k$ diagonal blocks. It follows that the block CMV matrix can be decomposed as a product of a unitary lower k-Hessenberg matrix multiplied by a unitary upper k-Hessenberg matrix.

2. An embedding technique which for a given triple (L, F, R) associated with A makes it possible to construct a larger matrix $\widehat{A} \in \mathbb{C}^{(n+k) \times (n+k)}$ which is still unitary plus rank-k and it can be factored as $\widehat{A} = \widehat{L} \cdot \widehat{F} \cdot \widehat{R}$, where $\widehat{L}$ is

the product of k unitary lower Hessenberg matrices, $\widehat{R}$ is the product of k unitary upper Hessenberg matrices, and the middle factor $\widehat{F}$ is unitary block diagonal plus rank-k with some additional properties.

3. A theoretical result which provides conditions under which a matrix specified in the form $\widehat{A} = \widehat{L} \cdot \widehat{F} \cdot \widehat{R}$ 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 $\widehat{A}$ into the LFR decomposition of an upper Hessenberg matrix $\widetilde{A}$ unitarily similar to $\widehat{A}$. The final representation of $\widetilde{A}$ thus involves $O(nk)$ data storage consisting of $O(k)$ vectors of length n and $O(nk)$ Givens rotations. The reduction to Hessenberg form turns out to have the same asymptotic complexity as eigensolvers for unitary plus low rank matrices and, furthermore, this representation is suited to be used directly by the fast eigensolver for unitary plus low rank matrices developed in [9].

The paper is organized as follows. In section 2 we introduce the LFR representations of unitary plus rank-k matrices by devising fast algorithms for transforming a matrix A into its LFR format provided that A belongs to some special classes. In section 3 we investigate the properties of LFR representations of unitary plus rank-k Hessenberg matrices and we describe a suitable technique to embed the matrix A into a larger matrix $\widehat{A}$ by maintaining its structural properties. In section 4 we present our algorithm which modifies the LFR representation of $\widehat{A}$ 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. The LFR format of unitary plus rank-k matrices. In this section we introduce a suitable compressed representation of unitary plus rank-k matrices which can be exploited for the design of fast Hessenberg reduction algorithms. We need some preliminary definitions.

A key role is played by generalized Hessenberg factors.

DEFINITION 2.1. *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 nonzero, that is, $h_{j+k,j} \neq 0$ ($h_{j,j+k} \neq 0$), $1 \leq j \leq m - k$, then the matrix is called* proper. *A matrix which is simultaneously k-lower and k-upper Hessenberg is called k-banded.*

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

Note that k-lower (upper) Hessenberg matrices can be obtained as the product of k matrices with the lower (upper) Hessenberg structure, and that unitary block Hessenberg matrices with blocks of size k are (nonproper) k-Hessenberg matrices.

DEFINITION 2.2. *A unitary plus rank-k matrix $A \in \mathbb{C}^{n \times n}$ can be represented in the LFR format if there is a triple (L, F, R) of matrices such that:*
1. *$A = LFR$;*
2. *$L \in \mathbb{C}^{n \times n}$ is a unitary k-lower Hessenberg matrix;*
3. *$R \in \mathbb{C}^{n \times n}$ is a unitary k-upper Hessenberg matrix;*
4. *$F = Q + [I_k, 0]^T Z^H \in \mathbb{C}^{n \times n}$ is a unitary plus rank-k matrix, where Q is a*

block diagonal unitary matrix of the form

$$Q = \left[\begin{array}{c|c} I_k & \\ \hline & \hat{Q} \end{array}\right]$$

with $\hat{Q}$ unitary Hessenberg and $Z \in \mathbb{C}^{n \times k}$.

CMV matrices [20] are pentadiagonal unitary matrices with a particular staircase structure of the nonzero elements. A block analogue of the CMV form of unitary matrices has been introduced in [3, 18].

DEFINITION 2.3 (block CMV). *A unitary matrix $G \in \mathbb{C}^{n \times n}$ is said to be* CMV structured with block size k *if there exist $k \times k$ nonsingular matrices R_i and L_i, respectively, upper and lower triangular, such that*

(2.1)
$$G = \begin{bmatrix} \times & \times & L_3 & & & & \\ R_1 & \times & \times & & & & \\ & \times & \times & \times & L_5 & & \\ & R_2 & \times & \times & \times & & \\ & & & \times & \times & & \\ & & & R_4 & \times & \ddots & \\ & & & & \ddots & \ddots \end{bmatrix}$$

or

$$G = \begin{bmatrix} \times & L_2 & & & & & \\ \times & \times & \times & L_4 & & & \\ R_1 & \times & \times & \times & & & \\ & & \times & \times & \times & L_6 & \\ & & R_3 & \times & \times & \times & \ddots \\ & & & & \ddots & \ddots & \ddots \end{bmatrix},$$

where the symbol $\times$ has been used to identify (possibly) nonzero $k \times k$ blocks.

Block CMV matrices are associated with matrix orthogonal polynomials on the unit circle and the structure of the matrix depends on the choice of the starting basis of the set of matrix polynomials to be orthogonalized. In particular, G fits the block structure shown in Definition 2.3 if $\left\{ I_k, zI_k, z^{-1}I_k, \ldots \right\}$ or $\left\{ I_k, z^{-1}I_k, zI_k, \ldots \right\}$ are considered. In what follows for the sake of simplicity we always assume that G satisfies the block structure (2.1). Furthermore, 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 continues 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.

In the rest of this section we present some fast algorithms for computing the LFR format of a unitary plus rank-k matrix $A \in \mathbb{C}^{n \times n}$ specified as follows:
- $A = G + [I_k, 0]^T Z^H$, $Z \in \mathbb{C}^{n \times k}$, and G is unitary block CMV with block size $k < n$;
- $A = H + [I_k, 0]^T Z^H$, $Z \in \mathbb{C}^{n \times k}$, and H is unitary block upper Hessenberg with block size $k < n$;
- $A = D + UV^H$, $U, V \in \mathbb{C}^{n \times k}$, and D is unitary block diagonal with block size $k < n$.

As pointed out in the introduction, these three cases cover the most interesting structures of low rank perturbation of unitary matrices. In the general case of unitary matrices, where the spectral factorization of the unitary part is not known or the unitary matrix cannot be represented in terms of a linear number of parameters, the retrieval of the eigenvalues—even only of the unitary part—requires $O(n^3)$ flops.

In the following sections we investigate the above three cases.

2.1. Small rank modifications of unitary block CMV matrices. Notice that a matrix in CMV form with blocks of size k is, in particular, $2k$-banded, i.e., is simultaneously $2k$-lower Hessenberg and $2k$-upper Hessenberg, with a total of $4k+1$ possibly nonzeros diagonals. The CMV structure with blocks of size 1 has been proposed as a generalization of what the tridiagonal structure is for Hermitian matrices in [12] and [20]. A further analogy between the scalar and the block case is derived from the nullity theorem [17] that is here applied to unitary matrices.

LEMMA 2.4 (nullity theorem). *Let U be a unitary matrix of size n. Then*

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

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

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

From Lemma 2.4 applied to a block CMV structured matrix G of block size k we find that for $p > 0$

$$0 = \text{rank}\left(G(1:2pk, (2p+1)k+1:n)\right) = \text{rank}\left(G(2pk+1:n, 1:(2p+1)k)\right) - k$$

which gives

$$\text{rank}\left(G(2pk+1:2(p+1)k, (2p-1)k+1:(2p+1)k)\right) = k.$$

Pictorially we are observing rank constraints on the following blocks:

$$G = \begin{bmatrix} \times & \times & L_3 & & & & \\ R_1 & \times & \times & & & & \\ & \boxed{\times} & \boxed{\times} & \times & L_5 & & \\ & R_2 & \times & \times & \times & & \\ & & & \boxed{\times} & \boxed{\times} & & \\ & & & R_4 & \times & \ddots & \\ & & & & \ddots & \ddots & \end{bmatrix}$$

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

In the scalar case with $k = 1$ these conditions make it possible to find a factorization of the CMV matrix as the product of two block diagonal matrices usually referred to as the classical Schur parametrization [11]. 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 2.5 (*CMV factorization*). *Let G be a unitary CMV structured matrix with blocks of size k as defined in Definition 2.3, (2.1). Then G can be factored into two block diagonal unitary matrices $G = G_1 G_2$ of the form*

$$G_1 = \mathrm{diag}(G_{1,1}, \ldots, G_{1,s}), \qquad G_2 = \mathrm{diag}(I_k, G_{2,2}, \ldots, G_{2,s+1})$$

such that $G_{2,s+1}$ has k rows and columns and all the other blocks $G_{i,j}$ have $2k$ rows and columns and bandwidth k with both $G_{i,j}(k+1:2k, 1:k)$ and $G_{i,j}(1:k, k+1:2k)$ triangular matrices of full rank. Moreover, each matrix G 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 G we have

$$
\begin{bmatrix} Q_{1,1} & Q_{1,2} \\ R_{2,1} & Q_{2,2} \\ & & I \end{bmatrix}^H
\begin{bmatrix}
\times & \times & L_3 & & & \\
R_1 & \times & \times & & & \\
& \times & \times & \times & L_5 & \\
& R_2 & \times & \times & \times & \\
& & \times & \times & & \\
& & & R_4 & \times & \ddots \\
& & & & \ddots & \ddots
\end{bmatrix}
=
\begin{bmatrix}
\tilde{\times} & & & & \\
& \tilde{\times} & \tilde{\times} & & & \\
& \times & \times & \times & L_5 & \\
& R_2 & \times & \times & \times & \\
& & \times & \times & & \\
& & & R_4 & \times & \ddots \\
& & & & \ddots & \ddots
\end{bmatrix},
$$

where $\tilde{\times}$ 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 G being unitary, and that the $R_{2,1}$ block is nonsingular upper triangular since it inherits the properties of R_1.
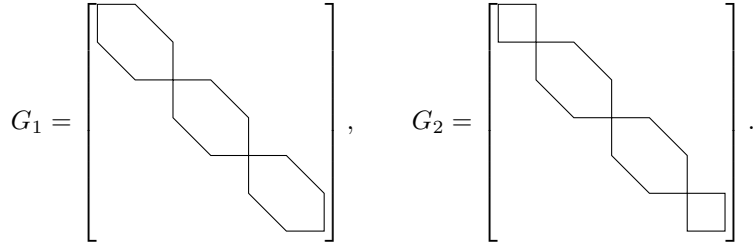
We can continue this process by computing the QR factorization of $\left[\begin{smallmatrix} \times \\ R_2 \end{smallmatrix}\right]$. Notice that, from the application of the nullity theorem 2.4 the block identified by $\left[\begin{smallmatrix} \times & \times \\ R_2 & \times \end{smallmatrix}\right]$ in the picture has rank at most k. This also holds for all the other blocks of the same kind. In particular, computing the QR factorization of the first k columns and left-multiplying by Q^H will also set to zero the block on the right of R_2. We will then get the following factorization:

$$
\begin{bmatrix} Q_{1,1} & Q_{1,2} \\ R_{2,1} & Q_{2,2} \\ & & Q_{3,3} & Q_{3,4} \\ & & R_{4,3} & Q_{4,4} \\ & & & & I \end{bmatrix}^H
\begin{bmatrix}
\times & \times & L_3 & & & \\
R_1 & \times & \times & & & \\
& \times & \times & \times & L_5 & \\
& R_2 & \times & \times & \times & \\
& & \times & \times & & \\
& & & R_4 & \times & \ddots \\
& & & & \ddots & \ddots
\end{bmatrix}
=
\begin{bmatrix}
\tilde{\times} & & & & \\
& \tilde{\times} & \tilde{\times} & & & \\
& \tilde{\times} & \tilde{\times} & & & \\
& & & \tilde{\times} & \tilde{\times} & \\
& & & \times & \times & \\
& & & R_4 & \times & \ddots \\
& & & & \ddots & \ddots
\end{bmatrix},
$$

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 G as the product of two unitary block diagonal matrices, that is $G = \widehat{G}_1 \widehat{G}_2$. This factorization can further be simplified by means of a block diagonal scaling $G = (\widehat{G}_1 D)(D^H \widehat{G}_2) = G_1 G_2$ with $D = \mathrm{diag}(D_1, \ldots, D_{2s})$, $D_{2j-1} = I_k$, and D_{2j} $k \times k$ unitary matrices determined so that the blocks $G_{i,j}$ are of bandwidth k, that is, the outermost blocks in G_1 and G_2 are triangular. 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 $L_3 = Q_{1,2}D_2(G_2)_{2,3} = R^H(G_2)_{2,3}$ it follows that the block of G_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 G_1 and G_2:

$$G_1 = \begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix} \quad , \qquad G_2 = \begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix}.$$

Now, let us assume that a matrix $A \in \mathbb{C}^{n \times n}$ is such that $A = G^T + [I_k, 0]^T Z^H$, $Z \in \mathbb{C}^{n \times k}$, and G is a unitary block CMV with block size $k < n$. By replacing G with its block diagonal factorization we obtain that $A = G_2^T(I_n + [I_k, 0]^T Z^H \bar{G}_1)G_1^T$. Since the left-hand and the right-hand side matrices are unitary k-banded it follows that they are both unitary k-Hessenberg matrices. Hence, we have the following.

THEOREM 2.6. *Let $A \in \mathbb{C}^{n \times n}$ be such that $A = G^T + [I_k, 0]^T Z^H$, $Z \in \mathbb{C}^{n \times k}$, and G is unitary block CMV with the block structure shown in (2.1). Then A can be represented in the LFR format as $A = G_2^T(I_n + [I_k, 0]^T \widehat{Z}^H)G_1^T$, where $L = G_2^T$, $R = G_1^T$, $G = G_1 G_2$ is the decomposition provided in Lemma 2.5, and $F = I_n + [I_k, 0]^T \tilde{Z}^H$, $\tilde{Z}^H = Z^H \bar{G}_1$.*

The overall cost of computing this condensed LFR representation of the unitary plus rank-k matrix A is $\mathcal{O}(nk^2)$ flops using $\mathcal{O}(nk)$ memory storage.

2.2. Small rank modifications of unitary block Hessenberg matrices. The class of perturbed unitary block Hessenberg matrices includes the celebrated block companion forms which are the basic tool in the construction of matrix linearizations of matrix polynomials. To be specific let $A \in \mathbb{C}^{n \times n}$ be a matrix such that $A = H + [I_k, 0]^T Z^H$, $Z \in \mathbb{C}^{n \times k}$, and H is a unitary block upper Hessenberg with block size $k < n$. A compressed LFR format of a matrix unitarily similar to A can be computed as follows. First of all we can suppose that all the subdiagonal blocks $H_{i+1,i}$, $1 \leq i \leq n/k$, are upper triangular. If not we consider the unitary block diagonal matrix P defined by $P = \text{blkdiag}[P_1, P_2, \ldots, P_{n/k}]$, where $P_i \in \mathbb{C}^{k \times k}$, $P_1 = I_k$, and $H_{i+1,i}P_i = P_{i+1}R_i$ is a QR decomposition of the matrix $H_{i+1,i}P_i$, $1 \leq i \leq n/k - 1$. Then the matrix $\tilde{A} = P^H A P$ is such that $\tilde{A} = \tilde{H} + [I_k, 0]^T \tilde{Z}^H$ and $\tilde{H}$ is unitary block upper Hessenberg with block size $k < n$ and $\tilde{H}_{i+1,i} = R_i$, $1 \leq i \leq n/k - 1$. Hence, the matrix $\tilde{H}$ is k-upper Hessenberg and therefore the factorization $\tilde{A} = I_n(I_n + [I_k, 0]^T \widehat{Z}^H)\tilde{H}$ gives a suitable LFR representation of $\tilde{A}$. Summing up we have the following.

THEOREM 2.7. *Let $A \in \mathbb{C}^{n \times n}$ be such that $A = H + [I_k, 0]^T Z^H$, $Z \in \mathbb{C}^{n \times k}$, and H is unitary block upper Hessenberg with block size $k < n$. Then there exists a unitary block diagonal matrix $P = \text{blkdiag}\left[P_1, P_2, \ldots, P_{n/k}\right]$, $P_i \in \mathbb{C}^{k \times k}$, $P_1 = I_k$ such that $\tilde{A} = P^H A P$ can be represented in the LFR format as $\tilde{A} = I_n(I_n + [I_k, 0]^T \widehat{Z}^H)\tilde{H}$, where $L = I_n$, $R = \tilde{G} = P^H G P$, and $F = I_n + [I_k, 0]^T \widehat{Z}^H$, $\widehat{Z}^H = Z^H \tilde{H}^H$.*

The overall cost of computing this condensed LFR representation of the unitary plus rank-k matrix A is $\mathcal{O}(nk^2)$ flops using $\mathcal{O}(nk)$ memory storage.

2.3. Small rank modifications of unitary block diagonal matrices. The unitary block diagonal matrix reduces to a unitary diagonal matrix up to a similarity transformation which can be performed within $O(nk^2)$ operations. The interest toward the properties of block CMV matrices is renewed in [18] 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 [18].

THEOREM 2.8. *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. Then, there exists a unitary matrix P such that $G = PDP^H$ is CMV structured with block size k and the block structure shown in Definition 2.3 and $PU = (e_1 \otimes I_k)U_1$ for some $U_1 \in \mathbb{C}^{k \times k}$. The matrices P, G, and U_1 can be computed with $O(n^2 k)$ operations.*

By applying Theorem 2.8 to the matrix pair (D^H, U) we find that there exists a unitary matrix P such that $G = PD^H P^H$ is CMV structured with block size k and $PU = (e_1 \otimes I_k)U_1$. In view of Lemma 2.5 this yields

$$P(D + UV^H)P^H = G^H + (e_1 \otimes I_k)U_1(PV)^H$$
$$= G_2^H(I + (e_1 \otimes I_k)Z^H)G_1^H$$

where $Z = G_1^H PVU_1^H \in \mathbb{C}^{n \times k}$. Since the left-hand and the right-hand side matrices are unitary k-banded they are also unitary k-lower and upper Hessenberg matrices. In this way we obtain the next result.

THEOREM 2.9. *Let $A \in \mathbb{C}^{n \times n}$ be such that $A = D + UV^H$ with $U, V \in \mathbb{C}^{n \times k}$, and D unitary diagonal. Then there exists a unitary matrix $P \in \mathbb{C}^{n \times n}$ such that $G = PDP^H$ has the block CMV structure shown in Definition 2.3 and $PU = (e_1 \otimes I_k)U_1$ for some $U_1 \in \mathbb{C}^{k \times k}$. Moreover, $\widetilde{A} = PAP^H$ can be represented in the LFR format as $\widetilde{A} = G_2^H(I_n + [I_k, 0]^T Z^H)\widetilde{G}_1^H$, where $L = G_2^H$, $R = G_1^H$, $PDP^H = G = G_1 G_2$ is the factorization of G provided in Lemma 2.5, and $F = I_n + [I_k, 0]^T Z^H$, $Z^H = U_1(PV)^H$.*

The overall cost of computing this condensed LFR representation of the unitary plus rank-k matrix A is $\mathcal{O}(n^2 k)$ flops using $\mathcal{O}(nk)$ 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 a unitary k-lower Hessenberg matrix, R is a unitary k-upper Hessenberg matrix, and the middle factor F is unitary plus rank-k, specifies a matrix in Hessenberg form. In section 4 we will discuss the chasing algorithm for reducing, by unitary similarity, a matrix of the form $L(I + (e_1 \otimes I_k)Z^H)R$ to Hessenberg form maintaining the factorization and enforcing the properness of the factor L to avoid breakdown of the subsequent QR iterations.

An important property of any unitary upper Hessenberg matrix $H \in \mathbb{C}^{m \times m}$ is that it can be represented as a 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

$$G_\ell = \begin{bmatrix} \alpha_\ell & \beta_\ell \\ -\beta_\ell & \bar{\alpha}_\ell \end{bmatrix}, |\alpha_\ell|^2 + \beta_\ell^2 = 1, \alpha_\ell, \in \mathbb{C}, \beta_\ell \in \mathbb{R}, \beta_\ell \geq 0,$$

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 $\alpha_\ell, \beta_\ell,$

$1 \leq \ell \leq m - 1$, and θ_m. The same representation also extends to unitary k-upper Hessenberg matrices since any matrix of this kind can be 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. Lower unitary Hessenberg matrices can be parametrized similarly as $H = \mathcal{G}_{m-1}\mathcal{G}_{m-2}\cdots\mathcal{G}_1\mathcal{D}_m$.

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. In fact, only working with an enlarged matrix do we have the guarantee that if F is in Hessenberg form A also has the same structure as will be clarified by Theorem 3.2. The embedding is also important for the bulge chasing algorithm as we explain in the next section. The following result is first proved in [9] and here specialized to a matrix of the form determined in Theorems 2.6, 2.7, and 2.9.

THEOREM 3.1. *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 $\widehat{U} \in \mathbb{C}^{m \times m}$, $m = n + k$, be defined as*

$$\widehat{U} = I_m - \begin{bmatrix} Q \\ -I_k \end{bmatrix} \begin{bmatrix} Q \\ -I_k \end{bmatrix}^H.$$

Then it holds that
  1. *$\widehat{U}$ is unitary;*
  2. *the matrix $\widehat{A} \in \mathbb{C}^{m \times m}$ given by*

$$\widehat{A} = \begin{bmatrix} L & \\ & I_k \end{bmatrix} \left(\widehat{U} + \left(\begin{bmatrix} G^H \\ 0 \end{bmatrix} + \begin{bmatrix} Q \\ -I_k \end{bmatrix} \right) \begin{bmatrix} Q \\ 0 \end{bmatrix}^H \right) \begin{bmatrix} R & \\ & I_k \end{bmatrix}$$

  satisfies

$$\widehat{A} = \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix}, \quad B \in \mathbb{C}^{n \times k}.$$

Proof. Property 1 follows by direct calculations from

$$\begin{bmatrix} Q \\ -I_k \end{bmatrix}^H \begin{bmatrix} Q \\ -I_k \end{bmatrix} = 2I_k.$$

For property 2 we find that

$$\widehat{U} + \left(\begin{bmatrix} G^H \\ 0 \end{bmatrix} + \begin{bmatrix} Q \\ -I_k \end{bmatrix} \right) \begin{bmatrix} Q \\ 0 \end{bmatrix}^H = \begin{bmatrix} I_n & Q \\ & 0_k \end{bmatrix} + \begin{bmatrix} I_k \\ 0 \end{bmatrix} \begin{bmatrix} Z \\ 0 \end{bmatrix}^H. \qquad \square$$

The unitary matrices L and R given in Theorems 2.6, 2.7, and 2.9 are k-Hessenberg matrices. The same clearly holds for the larger matrices $\mathrm{diag}(L, I_k)$ and $\mathrm{diag}(R, I_k)$ occurring in the factorization of $\widehat{A}$. The next result is the main contribution of this section and it provides conditions under which a matrix specified as a product $L \cdot \tilde{F} \cdot R$, where L is a unitary k-lower Hessenberg matrix R is a unitary k-upper Hessenberg matrix, and $\tilde{F}$ is a unitary matrix plus a rank-k correction, is in Hessenberg form.

In fact, once we apply the embedding described by Theorem 3.1 to $A = L(I + (e_1 \otimes I_k)Z^H)R$, the matrix obtained, $\widehat{A}$, is no longer in the LFR format since the middle factor is not in the prescribed format required by Definition 2.2. Moreover $\widehat{L} = L \oplus I_k$ is not a proper matrix, making implicit QR iterations subject to breakdown.

THEOREM 3.2. *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*

$$Q = \left[\begin{array}{c|c} I_k & \\ \hline & \hat{Q} \end{array} \right]$$

with $\hat{Q}$ $n \times n$ unitary upper Hessenberg. Let $F = Q + [I_k, 0]^T Z^H$ with $Z \in \mathbb{C}^{m \times k}$. Suppose that the matrix $\widehat{A} = LFR$ satisfies the block structure

$$\widehat{A} = \left[\begin{array}{cc} A & * \\ 0_{k,n} & 0_{k,k} \end{array} \right].$$

Then $\widehat{A}$ is an upper Hessenberg matrix.

Proof. We have that since L is unitary $L^H \widehat{A} = FR$. Note that $M = FR$ is $(k+1)$-upper Hessenberg since the rank-k correction is hidden in the first k rows and $Q \cdot R$ is $(k+1)$-upper Hessenberg. Because of the null blocks of $\widehat{A}$ we have

$$L^H(k+1:m, 1:n)A = M(k+1:m, 1:n).$$

From the properness of L we get that $L^H(k+1:m, 1:n)$ is nonsingular and upper triangular, so $A = (L^H(k+1:m, 1:n))^{-1}M(k+1:m, 1:n)$. The theorem is proven observing that $M(k=1:m, 1:n)$ is an upper Hessenberg matrix. □

4. The bulge chasing algorithm. In this section we present a bulge chasing algorithm relying upon Theorem 3.2 to compute the Hessenberg reduction of the matrix $\widehat{A}$ given as in Theorem 3.1, i.e., the embedding of $A = L(I + (e_1 \otimes I_k)Z^H)R$. We recall that Q and G are the factors of the economic QR factorization of Z.

Let us set

$$X = \left[\begin{array}{c} Q \\ -I_k \end{array} \right], \quad Y = \left[\begin{array}{c} G^H \\ 0 \end{array} \right] + X, \quad W = \left[\begin{array}{c} Q \\ 0 \end{array} \right],$$

so that we have

$$(4.1) \qquad \widehat{A} = \left[\begin{array}{cc} L & \\ & I_k \end{array} \right] \left(\widehat{U} + YW^H \right) \left[\begin{array}{cc} R & \\ & I_k \end{array} \right], \quad \widehat{U} = I_m - XX^H.$$

Observe that $X(k+1:m, :) = Y(k+1:m, :)$ and, moreover, $Y(n+1:m, :) = -I_k$ which implies $\mathrm{rank}(Y) = k$. In the preprocessing phase we initialize

$$L_0 := \left[\begin{array}{cc} L & \\ & I_k \end{array} \right], \quad R_0 := \left[\begin{array}{cc} R & \\ & I_k \end{array} \right], \quad X_0 := X, \quad Y_0 := Y, \quad W_0 := W.$$

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. In the cases considered in this paper, we rely on the additional structure of L_0, namely, that L_0 is also k-banded as we can observe from Theorems 2.6, 2.7 and 2.9.

In this section we make use of the following technical result.

LEMMA 4.1. *Let $B \in \mathbb{C}^{n \times n}$ be a unitary k-lower (k-upper) Hessenberg matrix. Let $H \in \mathbb{C}^{n \times n}$, be a unitary Hessenberg obtained as a sequence of ascending or descending Givens transformations acting on two consecutive rows, i.e., $H = \mathcal{G}_{n-1}\mathcal{G}_{n-2}, \cdots \mathcal{G}_1$ if H is lower Hessenberg or $H = \mathcal{G}_1 \mathcal{G}_2 \cdots \mathcal{G}_{n-1}$ if H is upper Hessenberg. Then, there exist a unitary k-lower (or k-upper) Hessenberg matrix $\tilde{B}$ (with the same orientation as B) and a unitary Hessenberg matrix $\tilde{H}$ such that $HB = \tilde{B}\tilde{H}$, where*

- $\tilde{H} = \begin{bmatrix} I_k & \\ & \hat{H} \end{bmatrix}$ *if B is k-lower Hessenberg,*

- $\tilde{H} = \begin{bmatrix} \hat{H} & \\ & I_k \end{bmatrix}$ *if B is k-upper Hessenberg,*

and $\hat{H}$ has the same orientation of H.

Proof. We prove the lemma only in the case H is lower Hessenberg and B is k-upper Hessenberg. We need to move each of the $n-1$ Givens rotations of H on the right of B. The first k Givens rotations of H, namely, $\mathcal{G}_1, \ldots, \mathcal{G}_k$, when applied to B do not destroy the k-lower Hessenberg structure of B, so that $\mathcal{G}_k\mathcal{G}_{k-1} \cdots \mathcal{G}_1 B = \hat{B}$ is still k-lower Hessenberg. When we apply $\mathcal{G}_{k+1}$ to $\hat{B}$ a bulge is produced in position $(k+2, 1)$, and we need to apply a rotation on the first two columns of $\mathcal{G}_{k+1}\hat{B}$ to remove the bulge, i.e., $\mathcal{G}_{k+1}\hat{B} = \hat{B}_1\tilde{\mathcal{G}}_1$; similarly, we can remove each of the remaining $n-k-1$ Givens rotations. At step i we have $\mathcal{G}_{k+i}\hat{B}_{i-1} = \hat{B}_i\tilde{\mathcal{G}}_i$. The last Givens $\mathcal{G}_{n-1}$ produces a bulge in position $(n, n-k-1)$ which can be removed by the rotation $\tilde{\mathcal{G}}_{n-k-1}$ acting on the columns $(n-k-1, n-k)$. We do not need to rotate the columns with indices between $n-k$ and n, so that

$$\tilde{H} = \tilde{\mathcal{G}}_{n-k-1} \cdots \tilde{\mathcal{G}}_2\tilde{\mathcal{G}}_1 = \begin{bmatrix} \hat{H} & \\ & I_k \end{bmatrix}.$$

We can similarly prove the remaining three cases.   □

The reduction of $\widehat{A} = \widehat{A}_0$ in Hessenberg form proceeds in three steps according to Theorem 3.2. The first two steps amount to determining a different representation of the same matrix $\widehat{A}_0$. In particular after these two steps the rank correction inside the brackets is confined to the first k-rows, while the L_0 factor on the left of the representation is substituted by a factor which is proper, and still with the lower k-Hessenberg structure. The third step is a bulge chasing scheme to complete the Hessenberg reduction.

1. (*preliminary phase* I) 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 taken as a k-lower Hessenberg proper matrix (see Lemma 2.4 of [9]). We can write

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

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

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

Notice that $\widehat{U}_1 := Q_0^H \widehat{U} R_0$ is a unitary $2k$-upper Hessenberg matrix. Indeed, we have that

$$\widehat{U}_1 = Q_0^H \widehat{U} Q_0 Q_0^H R_0 = (I_m - \hat{X}\hat{X}^H)Q_0^H R_0,$$

where $\hat{X} := Q_0^H X$. We have that

$$T_0 = Q_0^H Y = Q_0^H X + Q_0^H \begin{bmatrix} G^H \\ 0 \end{bmatrix},$$

hence,

$$\hat{X} = T_0 - Q_0^H \begin{bmatrix} G^H \\ 0 \end{bmatrix}.$$

Therefore $\hat{X}(2k+1 : m, :) = -Q_0^H(2k+1 : m, 1 : k)G^H = 0$ since $Q_0^H(2k+1 : m, 1 : k) = 0$. Hence, we have $\widehat{U}_1 = ((I_{2k} - \hat{X}(1 : 2k, :)\hat{X}^H(:, 1 : 2k)) \oplus I_{m-2k})Q_0^H R_0$ which, for the block diagonal structure of $I_m - \hat{X}\hat{X}^H$, turns out to be $2k$-upper Hessenberg.

2. (*preliminary phase* II) We can construct a unitary k-lower Hessenberg matrix P such that

(4.2)
$$\widehat{U}_1 P = \left[\begin{array}{c|c} I_k & 0 \\ \hline 0 & \hat{Q} \end{array} \right],$$

where $\hat{Q}$ is a unitary k-upper Hessenberg matrix. The matrix P can be constructed as follows: letting

$$P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix},$$

we set

$$\begin{bmatrix} P_{11} \\ P_{21} \end{bmatrix} = \widehat{U}_1^H(:, 1 : k).$$

Imposing (4.2) we have that $\widehat{U}_1(k+1 : m, 1 : n) = \hat{Q}P_{12}^H$, so $\hat{Q}$ and P_{12}^H can be determined as a Q and R factors of a QR factorization of $\widehat{U}_1(k+1 : m, 1 : n)$. Hence P_{12} can be taken lower triangular and the factor $\hat{Q}$ can be taken with the same lower profile as $\widehat{U}_1(k+1 : m, 1 : n)$, i.e., as a k-upper Hessenberg matrix. Finally P_{22} is obtained from the orthogonality relations as $P_{22} = \widehat{U}_1^H(n+1 : m, 1 : n)\hat{Q}$.

We obtain that

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

so that we can get back to $\widehat{A}_0$ as

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

As we observed at the beginning of section 4 the cases of interest for this paper are those where the initial matrix L_0 is simultaneously k-upper and k-lower Hessenberg, i.e., is k-banded. Then, applying k times Lemma 4.1, considering L_0 as a k-upper Hessenberg matrix, we can factorize $L_0 Q_0 = Q_1 L_1$, where Q_1 is a unitary k-lower Hessenberg matrix and

$$L_1 = \left[\begin{array}{c|c} I_k & \\ \hline & \hat{L}_1 \end{array} \right],$$

where $\hat{L}_1$ is a unitary k-upper Hessenberg matrix. It follows that

(4.3) $$\widehat{A}_0 = Q_1 \cdot (L_1 \widehat{U}_1 P + T_0 W_0^H R_0 P)P^H = Q_1(\widehat{U}_2 + (e_1 \otimes I_k)W_1^H)P^H,$$

where the matrix $\widehat{U}_2 := L_1 \widehat{U}_1 P$ satisfies

$$\widehat{U}_2 = \left[\begin{array}{c|c} I_k & \\ \hline & \tilde{U}_2 \end{array} \right],$$

where $\tilde{U}_2$ is a unitary $2k$-upper Hessenberg matrix, and $W_1 := P^H R_0^H W_0 \hat{T}_0^H$, where $\hat{T}_0 = T(1:k, 1:k)$. 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, since Q_0 is proper. From Lemma 2.4 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 by Theorem 3.2.

At the end of this step the enlarged matrix $\widehat{A}$ has been reduced to a product of a proper k-lower Hessenberg matrix Q_1 on the left, a unitary factor corrected in the first k rows, i.e., the term inside the brackets, and a k-upper Hessenberg matrix, i.e., P^H. Step 3 consists of the reduction of $\hat{U}_2$ to Hessenberg form so that the final matrix will be unitarily similar to $\widehat{A}$ and in the LFR format.

3. (*chasing phase*) We now need to work on the representation of $\widehat{A}_0$ in (4.3) to reduce the inner matrix $\widehat{U}_2$ to Hessenberg form by means of a bulge chasing procedure. Indeed Theorem 3.2 ensures that the matrix obtained will be in the LFR format and in Hessenberg form. These transformations will not affect the properness of the k-lower Hessenberg term on the left.

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

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

Then setting $G_1 = (I_{k+1} \oplus \mathcal{G}_1 \oplus I_{n-2k-1})$, we have

$$\widehat{A}_0 = Q_1 G_1^H (G_1 \widehat{U}_2 + (e_1 \otimes I_k) W_1^H) P^H.$$

The application of G_1^H on the right of the matrix 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 Q_1. This segment can be annihilated by a novel unitary upper Hessenberg matrix G_2 whose active part $\mathcal{G}_2 \in \mathbb{C}^{2k \times 2k}$ works on the left of $Q_1(:, k+2:3k+1)\mathcal{G}_1^H$ by acting on the rows of indices 2 through $2k+1$. We can then apply a similarity transformation to remove the bulge

$$G_2 \widehat{A}_0 G_2^H = Q_2 (G_1 \widehat{U}_2 + (e_1 \otimes I_k) W_1^H) P^H G_2^H,$$

where $Q_2 := G_2 Q_1 G_1^H$. The active part of G_2^H, the $2k \times 2k$ matrix $\mathcal{G}_2^H$, acts on the right of P^H producing 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 $3k+1$ of $P^H G_2^H$. Then, the matrix

$$\widehat{U}_2 \leftarrow G_1 \widehat{U}_2 (I_{k+1} \oplus \mathcal{G}_3^H \oplus I_{n-2k-1})$$

has a bulge on the rows of indices $2k+2$ through $4k+1$ which can be chased away by a sequence of $O(n/k)$ transformations having the same structure as above. Note that the rank correction of the unitary matrix inside the brackets is never affected by these transformations so that, at the end of the process, we have unitarily reduced A_0 to the LFR format in Definition 2.2. Also the zeros in the last k rows are preserved.

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

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

2. The cost of preliminary phase II 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 $O(nk^2)$ flops.

3. The dominant cost is the execution of the chasing steps. The zeroing of the subsubdiagonal entries costs $O(n \frac{n}{k} k^2) = O(n^2 k)$ flops.

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

5. Algorithmic details and 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 products of unitary Hessenberg matrices.

In preliminary phase I of the structured Hessenberg reduction scheme we first compute the full QR factorization of the matrix $Y_0 \in \mathbb{C}^{m \times k}$. The matrix Q_0^H turns out to be the product of k unitary upper Hessenberg matrices, each one representable as a descending sequence of $m - 1$ Givens transformations. Then we have to incorporate the unitary matrix $\mathcal{S} := I_{2k} - \hat{X}(1 : 2k, :)\hat{X}^H(:, 1 : 2k)$ on the right into the factored representations of Q_0^H and R_0. The unitary $2k \times 2k$ matrix $\mathcal{S}$ can always be represented as the product of at most $k(2k - 1)$ elementary unitary transformations of size 2×2. Once this factorization is computed, we have to add each of these single transformations, one by one, on the right to the factored representations of Q_0^H and R_0. This is accomplished by a sequence of turnover and fusion operations acting on the chains of elementary transformations in Q_0^H and R_0 (see [24] for the detailed description of these operations on elementary transformations).

At the beginning of preliminary phase II the matrix $\widehat{U}_1$ is a $2k$-upper Hessenberg matrix, and is essentially determined by the product of two unitary k-upper Hessenberg matrices that here we rename as $\widehat{U}_1 = \widehat{P}\widehat{Q}$. To reshape this factorization in the desired form we can apply k times a reasoning similar to the one done in Lemma 4.1 to move each elementary transformation of $\widehat{Q}$ on the left. In this way we find $\widehat{P}\widehat{Q} = \widetilde{Q}\widetilde{P}$, where

$$\widetilde{Q} = \left[\begin{array}{c|c} I_k & \\ \hline & \hat{Q} \end{array} \right]$$

is the matrix appearing in (4.2). Since $\widehat{Q}$ is formed by $O(nk)$ elementary transformations the reshaping costs $O(nk^2)$ flops. With a similar reasoning we can compute the representations of Q_1 and L_1, where Q_1 is k-lower Hessenberg and

$$L_1 = \begin{bmatrix} I_k & \\ & \hat{L}_1 \end{bmatrix}$$

with $\hat{L}_1$ unitary k-upper Hessenberg.

The chasing phase of the structured Hessenberg reduction scheme basically amounts to reducing the matrix $\widehat{U}_2 = L_1\widetilde{Q}$ into a matrix of the form

$$\left[\begin{array}{c|c} I_k & \\ \hline & \tilde{U}_2 \end{array} \right]$$

with $\tilde{U}_2$ $n \times n$ unitary Hessenberg. To be specific assume that $L_1 = L_{1,1} \cdots L_{1,k}$ and $\widetilde{Q} = \widetilde{Q}_1 \cdots \widetilde{Q}_k$, where $L_{1,j}$ and $\widetilde{Q}_j$ 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 $\widetilde{Q}_k, \ldots, \widetilde{Q}_1, L_{1,k}, \ldots, L_{1,1}$ are annihilated (in this order). Each transformation is moved on the left by finally creating a bulge in the leftmost factor $L_{1,1}$. This bulge is removed by applying a similarity transformation.

Let us consider in detail this first step. Let $L_{1,i} = \mathcal{G}_{k+1}^{(i)} \cdots \mathcal{G}_{m-1}^{(i)} D_m^{(i)}$ denote the Schur parametrization of $L_{1,i}$ and, similarly, let $\widetilde{Q}_i = \mathcal{H}_{k+1}^{(i)} \cdots \mathcal{H}_{m-1}^{(i)} E_m^{(i)}$ that of $\widetilde{Q}_i$. At this step we move left the first elementary transformations of each factor of the product $L_1 \widetilde{Q}$, for example, when moving the rotation $\mathcal{H}_{k+1}^{(k)}$ in front of L_1 the resulting transformation acts on rows $3k$ and $3k+1$ while some of the rotations in L_1 and $\widetilde{Q}$ have changed. The final situation is as follows:[1]

$$
\begin{aligned}
L_1 \widetilde{Q} &= (\mathcal{G}_{k+1}^{(1)} \cdots \hat{\mathcal{G}}_{m-1}^{(1)}) \cdots (\mathcal{G}_{k+1}^{(k)} \cdots \hat{\mathcal{G}}_{m-1}^{(k)}) (\mathcal{H}_{k+1}^{(1)} \cdots \hat{\mathcal{H}}_{m-1}^{(1)}) \cdots (\mathcal{H}_{k+1}^{(k)} \cdots \hat{\mathcal{H}}_{m-1}^{(k)}) D \\
&= \underbrace{(\tilde{\mathcal{H}}_{3k}^{(k)} \tilde{\mathcal{H}}_{3k-1}^{(k-1)} \cdots \tilde{\mathcal{H}}_{2k+1}^{(1)} \tilde{\mathcal{G}}_{2k}^{(k)} \cdots \tilde{\mathcal{G}}_{k+2}^{(2k)})}_{B} \underbrace{\tilde{\mathcal{G}}_{k+1}^{(1)} \tilde{L}_{1,1} \cdots \tilde{L}_{1,k} \hat{Q}_1 \cdots \hat{Q}_k D}_{\widehat{U}_2},
\end{aligned}
$$

where

$$
\tilde{L}_{1,j} = \tilde{\mathcal{G}}_{k+2}^{(j)} \cdots \tilde{\mathcal{G}}_{m-1}^{(j)} \quad \text{and} \quad \hat{Q}_j = \tilde{\mathcal{H}}_{k+2}^{(j)} \cdots \tilde{\mathcal{H}}_{m-1}^{(j)}.
$$

At this point we bring the bulge B on the left of Q_1 obtaining $Q_1 B = \widehat{B} \breve{Q}_1$. Substituting in (4.3) we obtain

$$
\widehat{A}_0 = \widehat{B} \breve{Q}_1 (\widehat{U}_2 + T_0 W_0^H R_0 P) P^H,
$$

where $\widehat{B} = \Gamma_{2k} \cdots \Gamma_2$ is the product of a sequence of elementary transformations in ascending order acting on rows $2 : 2k$. The bulge $\widehat{B}$ is removed by chasing an elementary transformation at a time. For example to remove Γ_{2k} we apply the similarity transformation $\Gamma_{2k}^H \widehat{B} \breve{Q}_1 (\widehat{U}_2 + T_0 W_0^H R_0 P) P^H \Gamma_{2k}$ 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 $O(nk \cdot k \cdot n/k) = O(n^2 k)$ flops. Note that the whole similarity transformation acts only on the first n rows leaving untouched the null rows at the bottom of $\widehat{A}$ in (4.1).

Numerical experiments have been performed to confirm the computational properties of the proposed method. Among the three cases considered in section 2, the last one, when the unitary part is block diagonal, is the most challenging since computing the starting LFR format costs $O(n^2 k)$ versus the $O(nk^2)$ flops sufficient for the first two cases. The CMV reduction of the input unitary diagonal plus rank-k matrix $D + UV^H$ is computed using the algorithm presented in [18] 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 Theorem 2.9. In the following tables we show the backward errors ϵ_P, ϵ_B, and ϵ_H generated by our procedure. These errors are defined as follows:

1. ϵ_P is the error computed at the end of the *preliminary phase* I *and* II. Given the matrix A of size n represented as in Theorem 3.1 we find the matrix $\widehat{A}$

---

[1] As observed, we can use only a unitary diagonal matrix to keep track of all the diagonal contributions.

TABLE 1

Backward errors for random matrices with $k = 2$. The results are the average over 10 random tests.

n	k	$\|A\|_2$	ϵ_P	ϵ_B	ϵ_H
32	2	$7.10e + 01$	$5.70e - 16$	$7.70e - 16$	$7.72e - 17$
64	2	$1.39e + 02$	$7.15e - 16$	$1.05e - 15$	$4.57e - 17$
128	2	$2.64e + 02$	$1.07e - 15$	$1.48e - 15$	$2.96e - 17$
256	2	$5.40e + 02$	$1.77e - 15$	$2.25e - 15$	$4.68e - 17$
512	2	$1.06e + 03$	$2.92e - 15$	$3.21e - 15$	$5.13e - 17$
1024	2	$2.10e + 03$	$5.05e - 15$	$5.43e - 15$	$4.47e - 17$

TABLE 2

Backward errors for random matrices of large norm with $k = 2$. The results are the average over 10 random tests.

n	k	$\|A\|_2$	ϵ_P	ϵ_B	ϵ_H
32	2	$3.34e + 06$	$5.82e - 16$	$8.15e - 16$	$2.58e - 21$
64	2	$1.01e + 08$	$8.20e - 16$	$1.11e - 15$	$3.57e - 22$
128	2	$3.14e + 09$	$1.18e - 15$	$1.48e - 15$	$8.33e - 24$
256	2	$1.01e + 11$	$1.67e - 15$	$2.05e - 15$	$3.08e - 25$
512	2	$3.17e + 12$	$3.09e - 15$	$3.33e - 15$	$9.89e - 25$
1024	2	$1.02e + 14$	$5.01e - 15$	$5.77e - 15$	$1.51e - 27$

of size $m = n + k$ obtained at the end of step 2. Denoting by $fl(\widehat{A})$ the computed matrix, the error is

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

2. ϵ_B is the classical *backward* error generated in the final step given by

$$\epsilon_B := \frac{\|H - Q fl(\widehat{A}) Q^H\|_2}{\|A\|_2},$$

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

3. ϵ_H is used to measure the Hessenberg structure of the matrix H. It is

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

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

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

The results of Tables 1, 2, 3, and 4 show that the proposed algorithm is numerically backward stable, and that the backward error ϵ_B might depend mildly on n as confirmed in the plot of Figure 1 where a comparison in terms of backward stability ϵ_B of our method with the algorithm proposed in [18] is shown. In particular, for $k = 2$ and n ranging from 32 to 1024, we report the backward error obtained by the two methods, together with the reference values $\sqrt{n}\,\epsilon$, where ϵ is the machine precision. We note that the two algorithms behave similarly with a slightly better performance of the method here proposed.

TABLE 3

Backward errors for random matrices with $k = 4$. The results are the average over 10 random tests.

n	k	$\|A\|_2$	ϵ_P	ϵ_B	ϵ_H
32	4	$8.09e + 01$	$7.55e - 16$	$1.07e - 15$	$4.54e - 16$
64	4	$1.58e + 02$	$9.69e - 16$	$1.31e - 15$	$1.01e - 15$
128	4	$2.96e + 02$	$1.26e - 15$	$1.82e - 15$	$6.55e - 16$
256	4	$5.65e + 02$	$2.18e - 15$	$2.68e - 15$	$2.06e - 15$
512	4	$1.10e + 03$	$3.74e - 15$	$4.04e - 15$	$9.33e - 16$
1024	4	$2.14e + 03$	$6.11e - 15$	$6.55e - 15$	$1.32e - 15$

TABLE 4

Backward errors for random matrices of large norm with $k = 4$. The results are the average over 10 random tests.

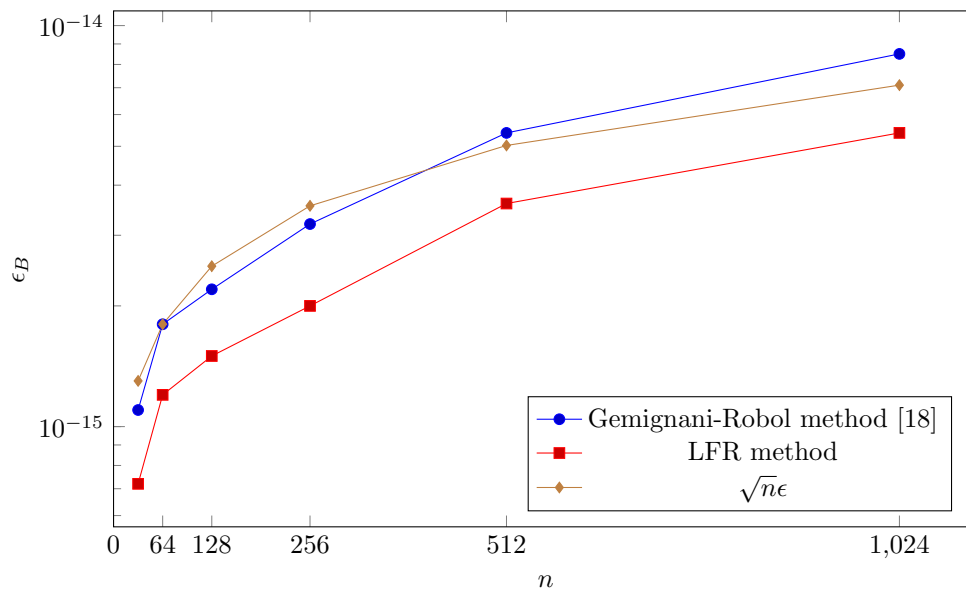| n	k	$\|A\|_2$	ϵ_P	ϵ_B	ϵ_H
32	4	$6.08e + 07$	$8.18e - 16$	$1.02e - 15$	$1.85e - 15$
64	4	$1.11e + 08$	$1.08e - 15$	$1.38e - 15$	$4.43e - 15$
128	4	$3.49e + 09$	$1.28e - 15$	$1.79e - 15$	$7.40e - 16$
256	4	$1.08e + 11$	$1.94e - 15$	$2.68e - 15$	$1.86e - 15$
512	4	$3.37e + 12$	$3.34e - 15$	$4.02e - 15$	$1.08e - 15$
1024	4	$1.05e + 14$	$5.94e - 15$	$6.56e - 15$	$5.88e - 16$

FIG. 1. *Comparison of the relative backward stability of the algorithm proposed in this paper, referred to as the "LFR method," and the one proposed in [18] on randomly generated diagonal plus rank-k matrices. We report the results for $k = 2$. The third line is for reference showing that the backward error behaves as $\sqrt{n}\,\epsilon$, ϵ being the machine precision.*

In the literature there are a few other papers dealing with the reduction of a rank-structured matrix into Hessenberg form. For example in [14] is proposed an algorithm of cost $O(n^2 k^2)$ flops which can be lowered to $O(n^2 k)$ in the case of unitary plus low rank matrices performing the required approximation phase only after $O(k)$ steps. However the algorithm makes use of the Givens-weight representation and at the moment no fast QR algorithm working with this representation is known. The
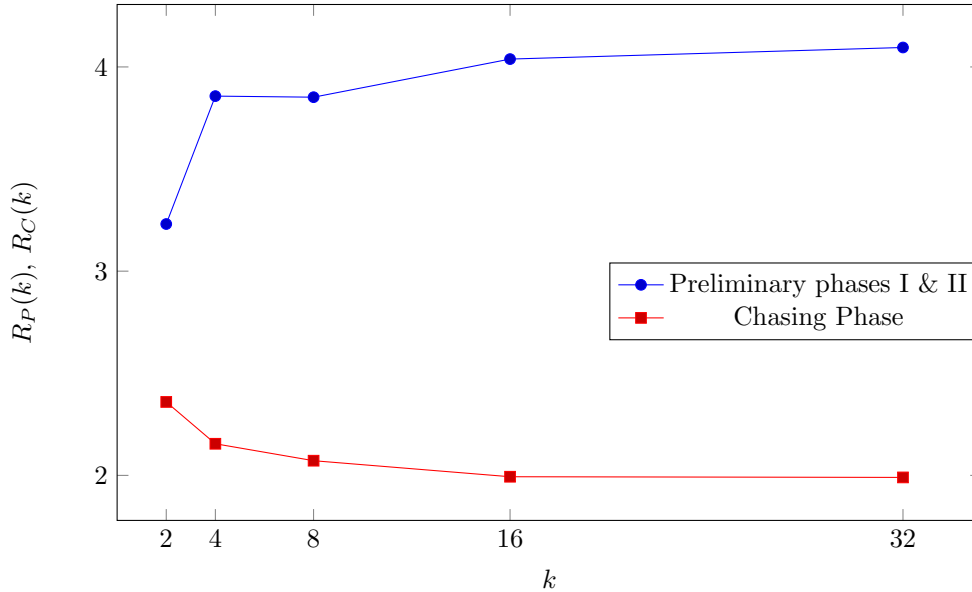
FIG. 2. *Dependence on k of the elapsed time. For random matrices of size 512 and k ranging from 1 to 32 we plot the ratio between elapsed times of the preliminary phases and of the chasing phase. We see that as k increases $R_P(k)$ approaches 4, while $R_C(k)$ tends to 2.*

algorithm proposed in [15] involves, on the other hand, $O(n^2 k^3)$ flops, and so it is not competitive when k is large.

For this reason we limited our comparison with the method proposed by Gemignani and Robol in [18] which requires $O(n^2 k)$ flops and is specialized to the unitary diagonal plus low rank matrices. However the Hessenberg matrix also obtained with that algorithm, which is based on the Givens-vector representation of the low rank part, is not directly exploitable by available fast eigensolvers.

In order to confirm the cost analysis of the algorithm of section 4 we have also performed experiments increasing the value of k and keeping n fixed, and increasing n and keeping k fixed. Denoting by $T_P(n, k)$ the time spent on the preliminary phases for a unitary diagonal matrices of size n corrected with a rank k matrix and by $T_C(N, k)$ the time required by the chasing steps on the matrix obtained after the preliminary phases, we consider the ratios $R_P(k) = T_P(n, 2k)/T_P(n, k)$ and $R_C(k) = T_C(n, 2k)/T_C(n, k)$. Since $T_P(n, k) = O(nk^2)$ we have that $R_P(k)$ should approach 4, and similarly $T_C(n, k) = O(n^2 k)$ means that $R_C(k)$ should approach 2. In Figure 2 we record the ratios $R_P(k)$ and $R_C(k)$ for a matrix of size 512 and k ranging from 2 to 32 confirming the quadratic growth of the cost of the preliminary phases $T_P(n, k)$, and the linear cost with respect to k of the chasing steps $T_C(n, k)$.

In Figure 3 we report a loglog plot of the time required by the algorithm, together with a reference line with slope proportional to 2, showing that the cost of the algorithm is indeed quadratic in n.

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
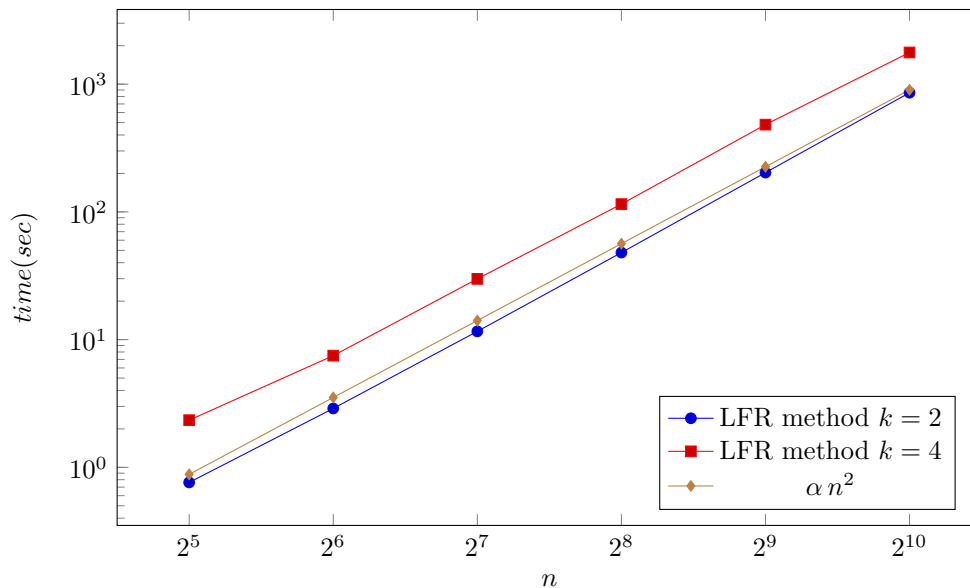
FIG. 3. *Loglog plot of the average over* 10 *random tests of the time required by the algorithm for values of n ranging from* 32 *to* 1024 *for* $k = 2$ *and* $k = 4$. *The line with slope* 2 *is plotted as reference, showing that the complexity in time is indeed* $O(n^2)$.

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 [9] 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. ARLINSKIĬ, *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*, Fundam. Algorithms FA13, SIAM, Philadelphia, 2018.

[5] J. AURENTZ, T. MACH, L. ROBOL, R. VANDEBRIL, AND D. S. WATKINS, *Fast and backward stable computation of eigenvalues and eigenvectors of matrix polynomials*, Math. Comp., 88 (2019), pp. 313–347, https://doi.org/10.1090/mcom/3338.

[6] A. P. AUSTIN, P. KRAVANJA, AND L. N. TREFETHEN, *Numerical algorithms based on analytic function values at roots of unity*, SIAM J. Numer. Anal., 52 (2014), pp. 1795–1821, https://doi.org/10.1137/130931035.

[7] R. BEVILACQUA, G. M. DEL CORSO, AND L. GEMIGNANI, *A CMV-based eigensolver for companion matrices*, SIAM J. Matrix Anal. Appl., 36 (2015), pp. 1046–1068.

[8] R. BEVILACQUA, G. M. DEL CORSO, AND L. GEMIGNANI, *A QR based approach for the nonlinear eigenvalue problem*, Rend. Semin. Mat. Univ. Politec. Torino, 76 (2018), pp. 77–87.

[9] R. BEVILACQUA, G. M. DEL CORSO, AND L. GEMIGNANI, *Fast QR iterations for unitary plus low rank matrices*, Numer. Math., 144 (2020), pp. 23–53, https://doi.org/10.1007/s00211-019-01080-4.

[10] D. A. Bini and L. Robol, *On a class of matrix pencils and ℓ-ifications equivalent to a given matrix polynomial*, Linear Algebra Appl., (2015), https://doi.org/10.1016/j.laa.2015.07.017.

[11] 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.

[12] 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.

[13] D. Damanik, J. Fillman, and R. Vance, *Dynamics of unitary operators*, J. Fractal Geom., 1 (2014), pp. 391–425, https://doi.org/10.4171/JFG/12.

[14] S. Delvaux and M. Van Barel, *A Hessenberg reduction algorithm for rank structured matrices*, SIAM J. Matrix Anal. Appl., 29 (2007), pp. 895–926, https://doi.org/10.1137/060658953.

[15] Y. Eidelman, I. Gohberg, and L. Gemignani, *On the fast reduction of a quasiseparable matrix to Hessenberg and tridiagonal forms*, Linear Algebra Appl., 420 (2007), pp. 86–101, https://doi.org/10.1016/j.laa.2006.06.028.

[16] Y. Eidelman, I. Gohberg, and I. Haimovici, *Separable Type Representations of Matrices and Fast Algorithms*. Vol. 1, vol. 234, 235 of Operator Theory: Advances and Applications, Birkhäuser, Basel, 2014.

[17] 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.

[18] 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.

[19] S. Ito and Y. Nakatsukasa, *Stable polefinding and rational least-squares fitting via eigenvalues*, Numer. Math., 139 (2018), pp. 633–682, https://doi.org/10.1007/s00211-018-0948-4.

[20] 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.

[21] 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.

[22] A. Sinap and W. Van Assche, *Orthogonal matrix polynomials and applications*, J. Comput. Appl. Math., 66 (1996), pp. 27–52, https://doi.org/10.1016/0377-0427(95)00193-X.

[23] W. Van Assche and A. Sinap, *Orthogonal matrix polynomials on the unit circle and applications*, in Orthogonal Polynomials on the Unit Circle: Theory and Applications (Madrid, 1994), Universidad Carlos III Madrid-Leganés, Madrid, 1994, pp. 159–171.

[24] 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.

[25] R. Vandebril, M. Van Barel, and N. Mastronardi, *Matrix Computations and Semiseparable Matrices*, Vol. 1, Johns Hopkins University Press, Baltimore, MD, 2008.