

# The Quadratic Arnoldi method for the solution of the quadratic eigenvalue problem

*Karl Meerbergen*

*Report TW 490, May 2007*



Katholieke Universiteit Leuven  
Department of Computer Science  
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

# The Quadratic Arnoldi method for the solution of the quadratic eigenvalue problem

*Karl Meerbergen*

*Report TW490, May 2007*

Department of Computer Science, K.U.Leuven

## **Abstract**

The Quadratic Arnoldi algorithm is an Arnoldi algorithm for the solution of the quadratic eigenvalue problem, that exploits the structure of the Krylov vectors. This allows us to reduce the memory requirements by about a half. The method is an alternative to the Second Order Arnoldi method (SOAR). In the SOAR method it is not clear how to perform an implicit restart. We discuss various choices of linearizations in  $\mathbb{L}_k$  and  $\mathbb{DL}$ . We also explain how to compute a partial Schur form of the underlying linearization with respect to the structure of the Schur vectors. We also formulate some open problems.

**Keywords :** Quadratic eigenvalue problem, Arnoldi method, SOAR, Schur decomposition

**AMS(MOS) Classification :** Primary : 15A18, 65F15, Secondary : 65G50

# THE QUADRATIC ARNOLDI METHOD FOR THE SOLUTION OF THE QUADRATIC EIGENVALUE PROBLEM

KARL MEERBERGEN\*

**Key words.** Arnoldi method, quadratic eigenvalue problem.

**AMS subject classification.** 15A18, 65F15

**Abstract.** The Quadratic Arnoldi algorithm is an Arnoldi algorithm for the solution of the quadratic eigenvalue problem, that exploits the structure of the Krylov vectors. This allows us to reduce the memory requirements by about a half. The method is an alternative to the Second Order Arnoldi method (SOAR). In the SOAR method it is not clear how to perform an implicit restart. We discuss various choices of linearizations in  $\mathbb{L}_1$  and  $\mathbb{DL}$ . We also explain how to compute a partial Schur form of the underlying linearization with respect to the structure of the Schur vectors. We also formulate some open problems.

**1. Introduction.** The goal is to solve the quadratic eigenvalue problem

$$Q(\lambda)u = 0 \quad \text{with} \quad Q(\lambda) = K + \lambda C + \lambda^2 M . \quad (1.1)$$

The matrices  $K$ ,  $-iC$ , and  $-M$  are the stiffness, damping, and mass matrices respectively and arise from the Fourier transformation of the spatial discretization by finite elements of the mechanical equation. They are large  $n \times n$  matrices and usually sparse. Equation (1.1) is solved when the engineer is interested in the eigenfrequencies (resonance frequencies) and damping properties of the mode shapes (i.e. the eigenvectors). Krylov methods for the solution of quadratic eigenvalue problems have been studied by Parlett and Chen [19], Saad [20], Mehrmann and Watkins [17]. The quadratic eigenvalue problem and solution methods are reviewed by Tisseur and Meerbergen [23].

Standard methods cannot be used directly to efficiently solve (1.1) because of the quadratic term in  $\lambda$ . Instead, (1.1) can be ‘linearized’ into a problem of the form

$$(A - \lambda B) \begin{pmatrix} \lambda u \\ u \end{pmatrix} = 0 . \quad (1.2)$$

Since we are interested in the eigenvalues near zero, we usually solve the inverted problem

$$S \begin{pmatrix} u \\ \theta u \end{pmatrix} = \theta \begin{pmatrix} u \\ \theta u \end{pmatrix} \quad (1.3)$$

with  $S = A^{-1}B$  and  $\theta = \lambda^{-1}$ . There are three disadvantages with the Arnoldi method : firstly, the doubling of the size of the problem increases the memory cost by a factor two ; secondly, the typical structure of the eigenvectors of the linearization is lost ; and finally, the Ritz values are computed from a small Hessenberg matrix and not from a small quadratic eigenvalue problem.

All these disadvantages can be solved by the SOAR method [3]. However, the SOAR method is not the preferred method for computing a Schur form of the linearization. As a consequence, classical implicit restarting [21] [18] is no longer possible.

---

\*K.U.Leuven, Department of Computer Science, 3001 Heverlee (Leuven), Belgium, [karl.meerbergen@cs.kuleuven.be](mailto:karl.meerbergen@cs.kuleuven.be)

In this document, we propose a method that is close to SOAR. We exploit the structure of the linear problem to divide the storage cost of the Arnoldi method roughly by a factor two ; we propose a locking procedure of converged Schur vectors that do keep the structure of the exact Schur vectors.

Both the SOAR method [3] and the Q-Arnoldi method compute the same subspace of the linearized problem. Q-Arnoldi performs a projection that does not produce a quadratic eigenvalue problem : the Ritz pairs are computed from the Arnoldi recurrence relation, which allows for the computation of a Schur form. Exploiting the Schur form of the linearization in the SOAR method for restarting purposing is not trivial and is, to date, an open question. See [14] for exploiting the Schur form in the inverse residual iteration and Jacobi-Davidson methods. The problem is that the Schur vectors, computed by the SOAR method, cannot, in general, be mapped in a Krylov recurrence relation.

Nevertheless, some results in this paper also have consequences to the SOAR method. We devote some time on the choice of linearization in  $\mathbb{L}_1$  and  $\mathbb{DL}$  [12], which is also useful for the SOAR method.

Note that  $B^{-1}A$  can be used as an alternative to  $A^{-1}B$ . In applications this is usually less effective than using  $A^{-1}B$  when the eigenvalues near zero are wanted. Many applications are not extremely large, i.e. smaller than 100,000 degrees of freedom, which allows the use of a direct linear system solver for applying  $A^{-1}$ .

The fact that the Krylov vectors have length  $2n$  instead of  $n$  may limit their practical use, especially when the Arnoldi method [1, 11, 2] is used when a large number of vectors need to be stored :  $k$  iterations of the Arnoldi method require the storage of the order of  $(2k + 2)n$  floating point numbers (real or complex). The Q-Arnoldi scheme exploits the structure of  $S$  to reduce the memory requirements to  $(k + 2)n$  floating point numbers. Similar tricks can be used for reducing the storage cost for partial reorthogonalization in the Lanczos method. Although the Lanczos method [8, 9] can be used to keep the storage requirements low, the Arnoldi method is usually preferred for eigenvalue computations. This memory reduction trick is also used in the SOAR trick.

The eigenvectors of (1.1) appear twice in the eigenvectors of (1.2). When the linearized problems are solved in a Krylov subspace, however, the two computed solutions are, in general, different. Therefore, we address which one of the two should be returned as an approximate solution vector of (1.1).

The applications we have in mind to solve arise from vibration problems. Such problems often have eigenvalues spread out along the imaginary axis, usually with the real parts on one side of the imaginary axis and often relatively small compared to the imaginary part. Since, usually, the eigenvalues with small imaginary part are desired, we use the shift-and-invert transformation with a pole near zero. This transformation has a cluster of eigenvalues near zero and a few out-liers, that may have large modulus, which has consequences to the use of the Q-Arnoldi algorithm. This will be discussed in the paper.

The paper is organized as follows. In §2, we introduce linearizations for (1.1). In §3, we review the Arnoldi method for (1.2) and present a modification of Arnoldi's method that saves memory. We call this the Q-Arnoldi algorithm, where Q stands for 'quadratic'. We discuss various issues including the choice of linearization, some thoughts on computations in finite precision arithmetic, and the choice of component of the Ritz vectors. Section 4 shows how to exploit the structure of (1.2) in the computation of Schur vectors. In §5 we show a numerical example from applications.

We conclude with some remarks in §6.

Throughout the paper, we use  $x^*$  to denote the Hermitian transpose,  $x^*y$  for the inner product of two vectors and  $\|x\| = \sqrt{x^*x}$  for the induced two-norm. We also use  $\|\cdot\|$  to denote the two-norm of a matrix. The matrix Frobenius norm is denoted by  $\|\cdot\|_F$ .

**2. Linearization via the first companion form.** By linearization, we mean the transformation of (1.1) into (1.2) by a suitable choice of  $A$ , and  $B$  so that there is a one-to-one correspondance between eigenpairs of (1.1) and the  $A - \lambda B$  pencil [22] [13] [12]. The linearization should be chosen so that the application of  $S = A^{-1}B$  is efficient and accurate. We therefore assume that  $A$  and so  $K$  are invertible.

Various choices of linearization can be used. Since  $K$  has no factor  $\lambda$  in (1.1), this suggests that  $K$  should appear in  $A$ . A straightforward choice is

$$A = \begin{bmatrix} D & \\ & K \end{bmatrix}, \quad B = \begin{bmatrix} & D \\ -M & -C \end{bmatrix}, \quad y = \begin{pmatrix} \lambda x \\ x \end{pmatrix} \quad (2.1)$$

where  $D$  can be any nonsingular matrix. It is easy to see that

$$S = A^{-1}B = \begin{bmatrix} & I \\ -K^{-1}M & -K^{-1}C \end{bmatrix} \quad (2.2)$$

from which  $D$  disappears.

LEMMA 2.1. *The pencil  $A - \lambda B$  with (2.1) is a linearization iff  $D$  is non-singular.*

*Proof.* If  $D$  is non-singular,  $A - \lambda B$  is a linearization. If  $D$  were singular,  $\lambda = 0$  is an eigenvalue of  $A - \lambda B$ , but not of (1.1). Also,  $\lambda = \infty$  is an eigenvalue even if it is not an eigenvalue of (1.1). In addition, it is no longer guaranteed that all eigenvectors have the form (1.2), so  $A - \lambda B$  is not a linearization.  $\square$

An alternative to (2.1) is the first companion form

$$B = \begin{bmatrix} -M & \\ & D \end{bmatrix}, \quad A = \begin{bmatrix} C & K \\ D & \end{bmatrix}, \quad y = \begin{pmatrix} \lambda x \\ x \end{pmatrix} \quad (2.3)$$

which also produces (2.2).

The matrix used by [17] for the skew-Hamiltonian Hamiltonian eigenvalue problem does not have the form (2.2).

The linearization can be chosen so that  $A$  and  $B$  respect special structure of  $K$ ,  $C$  and  $M$ . For example, if all matrices are symmetric and  $M$  is nonsingular, one could use  $D = -M$ . Although  $A^{-1}B$  is independent of  $D$ , it may help building the Krylov subspace more efficiently. We discuss this in more detail in §3.2.

**3. The Arnoldi method.** Let  $N = 2n$ . The Arnoldi method applied to  $S \in \mathbb{C}^{N \times N}$  and  $b \in \mathbb{C}^N$  produces the Krylov subspace

$$\mathcal{K}_k(S, b) = \text{span}\{b, Sb, S^2b, \dots, S^{k-1}b\}.$$

It computes the  $N \times k$  matrix  $\mathbf{V}_k = [\mathbf{v}_1, \dots, \mathbf{v}_k]$  of iteration vectors, the upper Hessenberg matrix  $H_k$  and the residual term  $\mathbf{v}_{k+1}\beta_k$  so that

$$\begin{aligned} S\mathbf{V}_k - \mathbf{V}_k H_k &= \mathbf{v}_{k+1}\beta_k e_k^*, \\ S\mathbf{V}_k - \mathbf{V}_{k+1} \underline{H}_k &= 0, \end{aligned} \quad (3.1)$$

where  $\mathbf{V}_{k+1}^* \mathbf{V}_{k+1} = I$ . Equation (3.1) is called the Arnoldi recurrence relation. An algorithm for computing  $\mathbf{V}_k$  and  $H_k$  is now given.

ALGORITHM 3.1 (Arnoldi method).

1. Set the initial vector  $\mathbf{v}_1$  so that  $\|\mathbf{v}_1\|_2 = 1$ .
  2. For  $j = 1, \dots, k$  do
    - 2.1. Compute  $\hat{\mathbf{v}}_j = S\mathbf{v}_j$ .
    - 2.2. Compute the Arnoldi coefficients  $h_j = \mathbf{V}_j^* \hat{\mathbf{v}}_j$ .
    - 2.3. Update  $\tilde{\mathbf{v}}_j = \hat{\mathbf{v}}_j - \mathbf{V}_j h_j$ .
    - 2.4. Get the scalar  $\beta_j = \|\tilde{\mathbf{v}}_j\|_2$  and compute  $\mathbf{v}_{j+1} = \tilde{\mathbf{v}}_j / \beta_j$ .
- End for

Steps 2.2–2.4 orthonormalize  $S\mathbf{v}_j$  against  $\mathbf{v}_1, \dots, \mathbf{v}_j$  into  $\mathbf{v}_{j+1}$ . The coefficients  $h_j$  form the  $j$ th column of  $H_j$  and  $\beta_j$  is the  $j+1, j$  element of  $\underline{H}_k$ . Roughly speaking,  $k$  iterations cost about  $2N(k+3)k$  flops, excluding the cost for Step 2.1, where 1 flop is the cost for an addition or a multiplication.

**3.1. The Q-Arnoldi algorithm.** We now discuss how we can make Algorithm 3.1 more efficient for  $S$  from (2.2). We decompose the  $j$ th Arnoldi vectors into

$$\mathbf{v}_j = \begin{pmatrix} v_j \\ w_j \end{pmatrix},$$

with  $v_j, w_j \in \mathbb{C}^n$ . The Arnoldi recurrence relation (3.1) for the linearization (2.1) can now be written as

$$\begin{bmatrix} I & \\ -K^{-1}M & -K^{-1}C \end{bmatrix} \begin{pmatrix} V_k \\ W_k \end{pmatrix} - \begin{pmatrix} V_k \\ W_k \end{pmatrix} H_k = \beta_k \begin{pmatrix} v_{k+1} \\ w_{k+1} \end{pmatrix} e_k^* \quad (3.2)$$

from which we deduce that

$$W_k = V_{k+1} \underline{H}_k. \quad (3.3)$$

This implies that we only have to store the vectors  $V_k$ ,  $v_{k+1}$  and  $w_{k+1}$  to evaluate the recurrence relation, which are  $(2+k)n$  floating point numbers. It results in an important reduction of the memory cost compared to Algorithm 3.1. The following algorithm implements this idea.

ALGORITHM 3.2 (Q-Arnoldi).

1. Let  $v_1$  and  $w_1$  be chosen so that  $\|v_1\|_2^2 + \|w_1\|_2^2 = 1$ .
2. For  $j = 1, \dots, k$  do
  - 2.1. Compute  $\hat{w}_j = -K^{-1}(Mv_j + Cw_j)$  and  $\hat{v}_j = w_j$ .
  - 2.2. Compute the Arnoldi coefficients

$$h_j = \begin{bmatrix} V_{j-1}^* \hat{v}_j + \underline{H}_{j-1}^* (V_j^* \hat{w}_j) \\ v_j^* \hat{v}_j + w_j^* \hat{w}_j \end{bmatrix}.$$

- 2.3. Update

$$\begin{aligned} \tilde{v}_j &= \hat{v}_j - V_j h_j, \\ \tilde{w}_j &= \hat{w}_j - [V_j \ w_j] \left( \begin{bmatrix} \underline{H}_{j-1} & 0 \\ 0 & 1 \end{bmatrix} h_j \right). \end{aligned}$$

- 2.4. Normalize  $v_{j+1} = \tilde{v}_j / \beta_j$  and  $w_{j+1} = \tilde{w}_j / \beta_j$  with  $\beta_j = (\|\tilde{v}_j\|^2 + \|\tilde{w}_j\|^2)^{1/2}$ .
- 2.5. Set the  $j$ th column of  $\underline{H}_j$  as  $[h_j^T \ \beta_j]^T$ .

End for

The difference with Algorithm 3.1 is only in Steps 2.2–2.3, where  $W_{j-1}$  is replaced by  $V_j \underline{H}_{j-1}$ . The cost for computing  $\underline{H}_{j-1}^* (V_j^* \hat{w}_j)$  in Step 2.2 is  $2(nj + j(j-1))$  flops, so that the total cost for Step 2.2 is  $2(2nj + n + j(j-1))$  flops. The cost for computing  $\hat{w}_j$  in Step 2.3 is  $2((j-1)j + n(j+1))$  and the computation of  $\tilde{v}_j$  requires  $2nj$  flops. Step 2.4 costs  $8n$  flops as for Algorithm 3.1. The total cost for Steps 2.2–2.4 is  $8nj + 12n + 4j(j-1)$ . For  $k$  iterations the cost is of the order of  $4nk^2 + 16nk + \frac{4}{3}(k-1)k(k+1)$  flops. Algorithm 3.1 requires  $4n(k+1)k$  flops. So, when  $k$  is significantly smaller than  $n$  (which is usually the case, otherwise the Arnoldi method is not suitable anyway), Steps 2.2–2.4 of Algorithms 3.1 and 3.2 have a cost of approximately  $4nk^2$ . Note that only  $v_1, \dots, v_{j+1}$  and  $w_{j+1}$  need to be stored on iteration  $j$  so that the memory requirements for the storage of the Arnoldi vectors for  $k$  iterations is limited to  $n(k+2)$  with the Q-Arnoldi method. The storage for the Arnoldi method is of the order of  $2n(k+1)$ . Although Q-Arnoldi is slightly more expensive in computation time, the extra cost is usually small compared to the computation of  $A^{-1}B\mathbf{v}_j$ .

**3.2. Other linearizations.** In this section, we study the use of other linearizations than the companion form in (2.3). Consider the vectorspace of linearizations [13] of the form

$$\mathbb{L}_1(Q) = \left\{ A - \lambda B : (A - \lambda B) \begin{pmatrix} \lambda \\ 1 \end{pmatrix} = \begin{pmatrix} \eta_1 Q(\lambda) \\ \eta_2 Q(\lambda) \end{pmatrix}, \eta_{1,2} \in \mathbb{C} \right\} \quad (3.4)$$

From (3.4), we have

$$A = \begin{bmatrix} A_{11} & \eta_1 K \\ A_{21} & \eta_2 K \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} -\eta_1 M & A_{11} - \eta_1 C \\ -\eta_2 M & A_{21} - \eta_2 C \end{bmatrix},$$

where  $A_{11}$ ,  $A_{21}$ ,  $\eta_1$  and  $\eta_2$  can be freely chosen.

**THEOREM 3.1.** *Let  $A$  and  $B$  be defined following (3.4) and let  $K$  be invertible. Then  $A - \lambda B$  is a linearization of (1.1) iff  $A$  is invertible. In addition, (2.2) holds and applying the Arnoldi method to  $A^{-1}B$  produces Arnoldi vectors with the structure (3.3).*

*Proof.* Let

$$S = A^{-1}B = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix}.$$

From  $B = AS$ , i.e.

$$\begin{bmatrix} -\eta_1 M & A_{11} - \eta_1 C \\ -\eta_2 M & A_{21} - \eta_2 C \end{bmatrix} = \begin{bmatrix} A_{11} & \eta_1 K \\ A_{21} & \eta_2 K \end{bmatrix} \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix}$$

it is easy to verify (2.2).

Formally,  $A$  does not need to be invertible for  $AS = B$  to be true. If both  $\eta_1 = \eta_2 = 0$ , then  $\det(A - \lambda B) = 0$  for all  $\lambda$ 's, so  $A - \lambda B$  is not a linearization. Suppose  $\eta_1 \neq 0$ . If we multiply the first block of  $A$  and  $B$  by  $\eta_2$  and the second block row by  $\eta_1$ , we find the pencil  $\tilde{A} - \tilde{B}$  with the same eigenvalues and eigenvectors, where

$$\tilde{A} = \begin{bmatrix} A_{11} & \eta_1 K \\ \eta_2 A_{11} - \eta_1 A_{21} & 0 \end{bmatrix} \quad \tilde{B} = \begin{bmatrix} -\eta_1 M & A_{11} - \eta_1 C \\ 0 & \eta_2 A_{11} - \eta_1 A_{21} \end{bmatrix}.$$

Since  $K$  is nonsingular,  $A$  can only be singular when

$$\eta_2 A_{11} - \eta_1 A_{21}$$

is singular. Following Lemma 2.1, the pencil  $\tilde{A} - \tilde{B}$  can only be a linearization when  $\eta_2 A_{11} - \eta_1 A_{21}$  is non-singular. In other words,  $A$  should be invertible for  $A - \lambda B$  being a linearization of (1.1).

Since  $A^{-1}B$  is the same matrix for all linearizations of this form, the Arnoldi method produces the same recurrence relation, which finishes the proof.  $\square$

One example that fits in this framework is the solution of palindromic eigenvalue problems [12]. Let  $K = M^T$  and  $C = C^T$ , then (1.1) is called T-palindromic. If  $(\lambda, x)$  is an eigenpair to (1.1) and the associated left eigenvector is  $y$  so that  $y^*Q(\lambda) = 0$ , then  $\lambda^{-1}$  also is an eigenvalue with (right) eigenvector  $\bar{y}$  and left eigenvector  $\bar{x}$ . In [12], methods are advocated that produce eigenvalue approximates that respect this spectral structure. They therefore introduce the linearization with  $B = -A^T$  in (3.4) of the form

$$A = \begin{bmatrix} K & K \\ C - M & K \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} -M & K - C \\ -M & -M \end{bmatrix}.$$

This corresponds to  $\eta_1 = \eta_2 = 1$ ,  $A_{11} = K$  and  $A_{21} = C - M$ .  $A$  is invertible when  $K$  and  $C - M - K$  are invertible, i.e. 0 and  $-1$  are no eigenvalues of (1.1). The linearized pencil  $A + \lambda(-B)$  is T-palindromic, since  $(-B) = A^T$ . Using  $A^{-1}B$  in the Arnoldi method, requires the inverse of  $K$  and  $C - M - K = C - K - K^T$ . However, note that the Ritz values from Arnoldi's method do not necessarily come in pairs of the form  $\lambda, \lambda^{-1}$ .

The generalization of the second companion form is given by

$$\mathbb{L}_2(Q) = \{A - \lambda B : (\lambda I \ I)(A - \lambda B) = (\tilde{\eta}_1 Q(\lambda) \ \tilde{\eta}_2 Q(\lambda)), \tilde{\eta}_{1,2} \in \mathbb{C}\}.$$

Similarly to  $\mathbb{L}_1$ , we can show that  $A$  and  $B$  take the form

$$A = \begin{bmatrix} A_{11} & A_{12} \\ \tilde{\eta}_1 K & \tilde{\eta}_2 K \end{bmatrix} \quad \text{with} \quad B = \begin{bmatrix} -\tilde{\eta}_1 M & -\tilde{\eta}_2 M \\ A_{11} - \tilde{\eta}_1 C & A_{12} - \tilde{\eta}_2 C \end{bmatrix}$$

and

$$BA^{-1} = \begin{bmatrix} 0 & -MK^{-1} \\ I & -CK^{-1} \end{bmatrix}.$$

Note that  $A^{-1}B$  depends on  $A_{11}$ ,  $A_{12}$ , and  $\tilde{\eta}_{1,2}$ .

The intersection of  $\mathbb{L}_1$  and  $\mathbb{L}_2$  is denoted by  $\mathbb{DL}(Q)$  [13]. Its general form is

$$A = \begin{bmatrix} \eta_1 C - \eta_2 M & \eta_1 K \\ \eta_1 K & \eta_2 K \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} -\eta_1 M & -\eta_2 M \\ -\eta_2 M & -\eta_2 C + \eta_1 K \end{bmatrix}$$

The very common cases for symmetric  $K$ ,  $C$ , and  $M$  are  $\eta_{1,2} = \{0, 1\}$  and  $\eta_{1,2} = \{1, 0\}$ . Note that  $A$  is invertible iff

$$\eta_1 \eta_2 C - \eta_2^2 M - \eta_1^2 K$$

is invertible, i.e.  $-\eta_2/\eta_1$  is not an eigenvalue of (1.1). Although working with pencils in  $\mathbb{DL}$  does not look very important for the Arnoldi method, it might be of interest when using the Lanczos method, see [19] [24]. The pseudo Lanczos method [19] is the Lanczos method applied to  $A^{-1}B$  using the  $B$  pseudo-inner product, i.e. the Lanczos vectors are orthogonal with respect to  $B$ , where  $A$  and  $B$  are chosen in  $\mathbb{DL}$ . Using the  $B$  inner product  $x^T B y$  rather than the Euclidean inner product  $x^* y$  produces a tridiagonal  $H_k$ .



### 3.3. Applying a shift.

**3.3.1. Shifting the quadratic equation.** The convergence of eigenvalues near  $\sigma$  can be improved [20, 2] by shifting the eigenvalue problem (1.1) into

$$(\tilde{K} + (\lambda - \sigma)\tilde{C} + (\lambda - \sigma)^2\tilde{M})u = 0, \quad (3.5)$$

with

$$\tilde{K} = K + \sigma C + \sigma^2 M, \quad \tilde{C} = C + 2\sigma M, \quad \tilde{M} = M.$$

Without loss of generality we can thus assume that  $\sigma = 0$ , by replacing  $K$ ,  $C$ , and  $M$  by  $\tilde{K}$ ,  $\tilde{C}$ , and  $\tilde{M}$  respectively, and  $\lambda - \sigma$  by  $\lambda$ .

**3.3.2. Shifting the linearization.** We can also shift (2.1) into

$$(A - \sigma B)^{-1}(A - \lambda B)y(\lambda) = 0,$$

The recurrence relation for  $(A - \sigma B)^{-1}B$  becomes

$$\begin{aligned} & \begin{bmatrix} A_{11} + \sigma\eta_1 M & \eta_1 K - \sigma A_{11} + \sigma\eta_1 C \\ A_{21} + \sigma\eta_2 M & \eta_2 K - \sigma A_{21} + \sigma\eta_2 C \end{bmatrix}^{-1} \begin{bmatrix} -\eta_1 M & A_{11} - \eta_1 C \\ -\eta_2 M & A_{21} - \eta_2 C \end{bmatrix} \begin{pmatrix} V_k \\ W_k \end{pmatrix} \\ & = \begin{pmatrix} V_{k+1} \\ W_{k+1} \end{pmatrix} \underline{H}_k. \end{aligned}$$

With  $\tilde{A}_{j1} = A_{j1} + \sigma\eta_j M$  this becomes

$$\begin{aligned} & \begin{bmatrix} \tilde{A}_{11} & \eta_1 \tilde{K} - \sigma \tilde{A}_{11} \\ \tilde{A}_{21} & \eta_2 \tilde{K} - \sigma \tilde{A}_{21} \end{bmatrix}^{-1} \begin{bmatrix} -\eta_1 M & \tilde{A}_{11} - \eta_1 \tilde{C} + \eta_1 \sigma M \\ -\eta_2 M & \tilde{A}_{21} - \eta_2 \tilde{C} + \eta_2 \sigma M \end{bmatrix} \begin{pmatrix} V_k \\ W_k \end{pmatrix} \\ & = \begin{pmatrix} V_{k+1} \\ W_{k+1} \end{pmatrix} \underline{H}_k. \end{aligned}$$

The pencil  $(A - \sigma B) - \mu B$  does not lie in  $\mathbb{L}_1(Q(\lambda - \sigma))$ . However, when we introduce  $Z_{k+1} = V_{k+1} - \sigma W_{k+1}$ , then we have

$$\begin{bmatrix} \tilde{A}_{11} & \eta_1 \tilde{K} \\ \tilde{A}_{21} & \eta_2 \tilde{K} \end{bmatrix}^{-1} \begin{bmatrix} -\eta_1 M & \tilde{A}_{11} - \eta_1 \tilde{C} \\ -\eta_2 M & \tilde{A}_{21} - \eta_2 \tilde{C} \end{bmatrix} \begin{pmatrix} Z_k \\ W_k \end{pmatrix} = \begin{pmatrix} Z_{k+1} \\ W_{k+1} \end{pmatrix} \underline{H}_k,$$

which is exactly the same as (3.2) with  $V_{k+1} = Z_{k+1}$ , where  $\tilde{K}$ ,  $\tilde{C}$  and  $M$  are related to the shifted problem (3.5). We compute  $W_k = Z_{k+1} \underline{H}_k$  and  $V_k = Z_{k+1} + \sigma W_k$ .

**3.4. Numerical stability.** A few words about numerical stability are in order. In this section, we perform a traditional rounding error analysis on the Algorithms 3.1 and 3.2.

The fact that we compute  $W_k$  from (3.3) changes the Arnoldi method in finite precision arithmetic. In this section, we show that, under certain conditions, the Q-Arnoldi method is as backward stable for the recurrence relation as the Arnoldi method. More specifically, we shall give bounds to

$$\tau_R = \|W_k - V_{k+1} \underline{H}_k\|_F \quad (3.6)$$

where  $\mathbf{V}_{k+1}$  and  $\underline{\mathbf{H}}_k$  are computed by either the Arnoldi method or the Q-Arnoldi method.

The reader is referred to Higham [7] for more details on computations with finite precision arithmetic. We denote by  $u$  the machine precision. We introduce the symbol  $\lesssim$  to denote

$$\alpha \lesssim \beta \Leftrightarrow \alpha \leq c\beta + \mathcal{O}(u)$$

where  $c$  is a  $u$  independent constant. We also bound  $\| |X| \cdot |Y| \|_F \leq \|X\|_F \cdot \|Y\|_F$ .

We define the error matrices  $F_j$  and  $G_j$  on the recurrence relation (3.1):

$$\hat{V}_j - V_{j+1}\underline{\mathbf{H}}_j = F_j, \quad (3.7a)$$

$$\hat{W}_j - W_{j+1}\underline{\mathbf{H}}_j = G_j. \quad (3.7b)$$

**3.5. Analysis for the Arnoldi algorithm.** We first show two lemma's that will help us to make a statement on the backward stability of the Arnoldi method.

LEMMA 3.2. *For all  $j = 1, \dots, k$  we have and*

$$\|v_{j+1}\beta_j - \tilde{v}_j\| \lesssim u\|\tilde{v}_j\| \lesssim u\|v_{j+1}\|\beta_j, \quad (3.8a)$$

$$\|w_{j+1}\beta_j - \tilde{w}_j\| \lesssim u\|\tilde{w}_j\| \lesssim u\|w_{j+1}\|\beta_j, \quad (3.8b)$$

$$\left\| \begin{pmatrix} v_{j+1} \\ w_{j+1} \end{pmatrix} \beta_j - \begin{pmatrix} \tilde{v}_j \\ \tilde{w}_j \end{pmatrix} \right\| \lesssim u \left\| \begin{pmatrix} \tilde{v}_j \\ \tilde{w}_j \end{pmatrix} \right\| \lesssim u\beta_j. \quad (3.8c)$$

We also have

$$\|V_j\|_F^2 + \|W_j\|_F^2 = j + \mathcal{O}(u). \quad (3.9)$$

*Proof.* The proofs of (3.8a), (3.8b) and (3.8c) follow from the fact that  $v_{j+1}$  and  $w_{j+1}$  are computed as  $v_{j+1} = \tilde{v}_j/\beta_j$  and  $w_{j+1} = \tilde{w}_j/\beta_j$ . The proof of (3.9) follows from Theorem 18.12 with  $m = 1$  in [7] for  $\|v_j\|^2 + \|w_j\|^2 = 1 + \mathcal{O}(u)$ .  $\square$

LEMMA 3.3. *In finite precision arithmetic, Algorithm 3.1 produces  $V_{j+1}$ ,  $W_{j+1}$  and  $\underline{\mathbf{H}}_j$  so that (3.7) holds with*

$$\|F_j\|_F \lesssim u\|V_{j+1}\| \cdot \|\underline{\mathbf{H}}_j\|_F \lesssim u\|V_{j+1}\|_F \|\underline{\mathbf{H}}_j\|_F, \quad (3.10a)$$

$$\|G_j\|_F \lesssim u\|W_{j+1}\| \cdot \|\underline{\mathbf{H}}_j\|_F \lesssim u\|W_{j+1}\|_F \|\underline{\mathbf{H}}_j\|_F, \quad (3.10b)$$

$$\left\| \begin{pmatrix} \hat{V}_j - V_{j+1}\underline{\mathbf{H}}_j \\ \hat{W}_j - W_{j+1}\underline{\mathbf{H}}_j \end{pmatrix} \right\|_F \lesssim u\|\underline{\mathbf{H}}_j\|_F. \quad (3.10c)$$

In addition, we have

$$\left\| h_j - \begin{pmatrix} V_j \\ W_j \end{pmatrix}^* \begin{pmatrix} \hat{v}_j \\ \hat{w}_j \end{pmatrix} \right\| \lesssim u \left\| \begin{pmatrix} \hat{v}_j \\ \hat{w}_j \end{pmatrix} \right\|. \quad (3.11)$$

*Proof.* Recall that we can omit constant factors from error bounds using the notation  $\lesssim$ . We first prove (3.10a). We define

$$\tilde{f}_j = \hat{v}_j - V_j h_j - \tilde{v}_j, \quad (3.12)$$

where  $\tilde{v}_j$  is computed in Step 2.3 of Algorithm 3.1. We have

$$\|\tilde{f}_j\| \lesssim u(\|\hat{v}_j\| + \| |V_j| \cdot |h_j| \|).$$

Denote by  $f_j$  the last column of  $F_j$ .

$$f_j = \hat{v}_j - V_j h_j - v_{j+1} \beta_j = \tilde{f}_j + (\tilde{v}_j - v_{j+1} \beta_j) . \quad (3.13)$$

where  $v_{j+1}$  and  $\beta_j$  are computed from Step 2.4. The application of Lemma 3.2 on Step 2.4 of Algorithm 3.1 readily produces

$$\|f_j\| \lesssim u(\|\hat{v}_j\| + \|V_j\| \cdot \|h_j\| + \|v_{j+1}\| \|\beta_j\|) \lesssim u \left\| |V_{j+1}| \cdot \begin{pmatrix} h_j \\ \beta_j \end{pmatrix} \right\| ,$$

using  $\|\hat{v}_j\| \lesssim \|V_j\| \|h_j\| + |v_{j+1}| \|\beta_j\|$ . Accumulating  $f_j$  in  $F_j = [f_1, \dots, f_j]$  proves (3.10a). The proofs for (3.10b) and (3.10c) are similar. We use (3.9) to bound (3.10c).

The proof of (3.11) readily follows from standard rounding error analysis.  $\square$

**3.6. Analysis for the Q-Arnoldi algorithm.** We define

$$\delta_j = \|V_j\|_F , \quad (3.14)$$

$$\gamma_j = \| [|V_j| \cdot \underline{H}_{j-1} \quad w_j] \|_F , \quad (3.15)$$

$$\tilde{\gamma}_{j+1} = \| [|V_j| \cdot \underline{H}_{j-1} \quad w_j \quad w_{j+1}] \|_F . \quad (3.16)$$

For the Q-Arnoldi algorithm,  $W_j z$  and  $\begin{pmatrix} V_j \\ W_j \end{pmatrix}^* \begin{pmatrix} \hat{v}_j \\ \hat{w}_j \end{pmatrix}$  are computed as

$$[V_j \underline{H}_{j-1} \quad w_j] z \quad \text{and} \quad \begin{pmatrix} \underline{H}_{j-1}^* (V_j^* \hat{w}_j) \\ w_j^* \hat{w}_j \end{pmatrix} + V_j^* \hat{v}_j \quad (3.17)$$

respectively. The componentwise errors on the computation of (3.17) are of the order

$$u [|V_j| \cdot \underline{H}_{j-1} \quad |w_j|] |z| \quad \text{and} \quad u \left( \frac{|\underline{H}_{j-1}|^* (|V_j|^* |\hat{w}_j|)}{|w_j|^* |\hat{w}_j|} \right) + |V_j|^* |\hat{v}_j| \quad (3.18)$$

respectively. The normwise error bounds are

$$u \gamma_j \|z\| \quad \text{and} \quad u(\gamma_j + \delta_j) \left\| \begin{pmatrix} \hat{v}_j \\ \hat{w}_j \end{pmatrix} \right\| \quad (3.19)$$

respectively.

We extend Lemma 3.3 to the Q-Arnoldi algorithm.

LEMMA 3.4. *In finite precision arithmetic, Algorithm 3.2 produces  $V_{j+1}$ ,  $W_{j+1}$  and  $\underline{H}_j$  so that (3.7) holds with*

$$\|F_j\|_F \lesssim u \delta_{j+1} \|\underline{H}_j\|_F , \quad (3.20a)$$

$$\|G_j\|_F \lesssim u \gamma_{j+1} \|\underline{H}_j\|_F , \quad (3.20b)$$

$$\left\| \begin{pmatrix} \hat{V}_j - V_{j+1} \underline{H}_j \\ \hat{W}_j - W_{j+1} \underline{H}_j \end{pmatrix} \right\|_F \lesssim u \sqrt{\gamma_{j+1}^2 + \delta_{j+1}^2} \|\underline{H}_j\|_F . \quad (3.20c)$$

*Proof.* Similar to the proof of (3.10a), we find for the componentwise analysis that

$$|F_j| = |W_j - V_{j+1} \underline{H}_j| \lesssim u |V_{j+1}| \cdot |\underline{H}_j| ,$$

which proves (3.20a).

We now prove (3.20b). We define

$$\tilde{g}_j = \hat{w}_j - [V_j \underline{H}_{j-1} \quad w_j] h_j - \tilde{w}_j \quad (3.21)$$

where  $\tilde{w}_j$  is computed in Step 2.3 of Algorithm 3.2. We have

$$\|\tilde{g}_j\| \lesssim u(\|\hat{w}_j\| + \|[|V_j| \cdot |\underline{H}_{j-1}| \quad |w_j|] |h_j|\|) .$$

Let

$$g_j = \hat{w}_j - W_j h_j - w_{j+1} \beta_j , \quad (3.22)$$

$$= \tilde{g}_j + (\tilde{w}_j - w_{j+1} \beta_j) + ([V_j \underline{H}_{j-1} \quad w_j] - W_j) h_j , \quad (3.23)$$

where  $w_{j+1}$  and  $\beta_j$  are computed from Step 2.4. Using Lemma 3.2, we find that

$$\begin{aligned} \|g_j\| &\lesssim u(\|\hat{w}_j\| + \|[|V_j| \cdot |\underline{H}_{j-1}| \quad |w_j|] |h_j|\| + \|w_{j+1}\| \beta_j + \|[|V_j| \cdot |\underline{H}_{j-1}| \quad 0] |h_j|\|) \\ &\lesssim u \left\| \left[ |V_j| \cdot |\underline{H}_{j-1}| \quad |w_j| \quad |w_{j+1}| \right] \begin{pmatrix} h_j \\ \beta_j \end{pmatrix} \right\| . \end{aligned}$$

Accumulating  $g_j$  in  $G_j = [g_1, \dots, g_j]$  and noting that  $\tilde{\gamma}_{j+1} \leq \gamma_{j+1}$  proves (3.20b). The proof for (3.20c) is similar.

□

Define

$$\xi_{j,\min} = \sqrt{\sigma_{\min}(\underline{H}_{j-1})^2 + 1} \quad (3.24)$$

$$\xi_{j,\max} = \sqrt{\sigma_{\max}(\underline{H}_{j-1})^2 + 1} . \quad (3.25)$$

Note that  $\xi_{j,\min} \leq 1 \leq \xi_{j,\max}$ . For the vibration problems we have in mind,  $S$  usually has very small eigenvalues, usually leading to  $\xi_{j,\min} \approx 1$ . The choice of pole  $\sigma$  (see §3.3) may influence the large singular values of  $S$ . By selecting the pole not too close to an eigenvalue (which is common practice when several eigenvalues are wanted [6] [15]),  $\xi_{j,\max}$  is not very large.

**THEOREM 3.5.** *In exact arithmetic,*

$$\sqrt{\gamma_j^2 + \delta_j^2} = \left\| \begin{pmatrix} V_{j-1} \\ |V_j| \cdot |\underline{H}_{j-1}| \end{pmatrix} \right\|_2 \leq \frac{\xi_{j,\max}}{\xi_{j,\min}} .$$

*Proof.* First,

$$\sigma_{\min} \left( \begin{pmatrix} I_{j-1} \\ \underline{H}_{j-1} \end{pmatrix} \right) = \lambda_{\min}^{1/2}(I + \underline{H}_{j-1}^* \underline{H}_{j-1}) = \sqrt{\sigma_{\min}(\underline{H}_{j-1})^2 + 1} = \xi_{j,\min} .$$

Similarly,

$$\sigma_{\max} \left( \begin{pmatrix} I_{j-1} \\ \underline{H}_{j-1} \end{pmatrix} \right) = \xi_{j,\max} .$$

From

$$\begin{pmatrix} V_{j-1} \\ V_j \underline{H}_{j-1} \end{pmatrix} = \begin{bmatrix} V_j & 0 \\ 0 & V_j \end{bmatrix} \begin{pmatrix} I_{j-1} \\ \underline{H}_{j-1} \end{pmatrix}$$

and

$$\left\| \begin{pmatrix} V_{j-1} \\ V_j \underline{H}_{j-1} \end{pmatrix} \right\|_2 = 1 ,$$

we have that

$$\xi_{j,\max}^{-1} \leq \|V_j\|_2 \leq \xi_{j,\min}^{-1} .$$

The proof follows from

$$\left\| \begin{bmatrix} V_j & 0 \\ 0 & V_j \end{bmatrix} \begin{pmatrix} \underline{L}_{j-1} \\ \underline{H}_{j-1} \end{pmatrix} \right\| \leq \xi_{j,\min}^{-1} \xi_{j,\max} .$$

□

The conclusion from this section, is that a loss of precision is possible in the computation of  $\underline{H}_k$  (see (3.11) and (3.19)) and the recurrence relation in the Q-Arnoldi process, when  $\xi_{k,\max}/\xi_{k,\min}$  is large.

**4. The solution of the quadratic eigenvalue problem.** To simplify, we write  $S$  in the form

$$S = \begin{bmatrix} & I \\ S_1 & S_2 \end{bmatrix} .$$

The solution of the quadratic eigenvalue problem by the shift-invert Arnoldi method is the objective of this section. For the computation of a number of eigenvalues of a non Hermitian linear problem, we usually compute a partial Schur form. The idea is that we want the computed Schur vectors to have the structure of the exact Schur vectors. It turns out that when we force the Schur vectors to satisfy this structure, we can also keep the structure of the Krylov vectors. In addition, implicit restarting also keeps the structure of the Krylov vectors.

We first introduce the notion of Q-Arnoldi triple.

**4.1. Definition and properties of Q-Arnoldi triples.** DEFINITION 4.1.  $\mathcal{Q} = \{V_{k+1}, \underline{H}_k, w_{k+1}\}$  is a Q-Arnoldi triple associated with  $S$  iff  $V_{k+1} \in \mathbb{C}^{n \times (k+1)}$ ,  $w_{k+1} \in \mathbb{C}^n$ ,  $\underline{H}_k \in \mathbb{C}^{(k+1) \times k}$  and for

$$\mathbf{V}_k = \begin{pmatrix} V_k \\ V_{k+1} \underline{H}_k \end{pmatrix} \quad \text{and} \quad \mathbf{V}_{k+1} = \begin{pmatrix} V_k & w_{k+1} \\ V_{k+1} \underline{H}_k & w_{k+1} \end{pmatrix} \quad (4.1)$$

1. the Arnoldi recurrence relation (3.1) holds
2. and the Arnoldi vectors are orthogonal:

$$\|I - \mathbf{V}_{k+1}^* \mathbf{V}_{k+1}\| = 0 . \quad (4.2)$$

DEFINITION 4.2. We denote by  $\mathbf{Q}_k(S)$  the set of all Q-Arnoldi triples associated with  $S$ .

**4.1.1. Inexact Q-Arnoldi triple.** In practice, we may allow a small error on (3.1) and (4.2) so that

$$S \mathbf{V}_k - \mathbf{V}_{k+1} \underline{H}_k = F_k , \quad (4.3)$$

and

$$\|I - \mathbf{V}_{k+1}^* \mathbf{V}_{k+1}\| = \gamma_k . \quad (4.4)$$

where  $F_k$  can be considered a backward error on  $S$ .

DEFINITION 4.3. *The set of inexact Q-Arnoldi triples*

$$\mathbf{Q}_k(S, \eta, \rho)$$

consists of  $\{V_{k+1}, \underline{H}_k, w_{k+1}\}$  that satisfy (4.3), (4.4) and (4.1) with  $\|F_k\| \leq \eta$  and  $\gamma_k \leq \rho$ .

**4.1.2. Transformations of (inexact) Q-Arnoldi triples.** DEFINITION 4.4.

Let  $Z$  be a full column rank  $k+1 \times p+1$  matrix of the form  $Z = \begin{pmatrix} Z_1 & z \\ 0 & \zeta \end{pmatrix}$  with  $Z_1$  an  $k \times p$  matrix and  $k \geq p$ . We define the transformation

$$\mathcal{T}_Z \{V_{k+1}, \underline{H}_k, w_{k+1}\} = \{V_{k+1}Z, Z^\dagger \underline{H}_k Z_1, V_{k+1} \underline{H}_k z + w_{k+1} \zeta\} ,$$

where  $Z^\dagger$  is the generalized inverse, i.e.  $Z^\dagger Z = I$ . The following theorem characterizes the transformation of an (inexact) Q-Arnoldi triple.

THEOREM 4.5. *Let  $\mathcal{Q} \in \mathbf{Q}_k(S, \eta, \rho)$ . Let  $\mathcal{T}_Z$  be a transformation as defined by Definition 4.4. If*

$$ZZ^\dagger \underline{H}_k Z_1 = \underline{H}_k Z_1$$

then

$$\mathcal{T}_Z \mathcal{Q} \in \mathbf{Q}_p(S, \eta \|Z_1\|, \|Z\|^2 \rho + \|I - Z^* Z\|) .$$

*Proof.* Let  $\mathcal{Q} = \{V_{k+1}, \underline{H}_k, w_{k+1}\} \in \mathbf{Q}_k(S, \eta, \rho)$ . Under the condition of the Theorem, the elements of  $\mathcal{T}_Z \mathbf{Q}_k(S)$  respect the structure (4.1). Multiplication of (4.3) on the right by  $Z_1$  proves that the error on the recurrence relation of the transformed triple is bounded by  $\|Z_1\| \eta$ . Finally, we have that

$$I - Z^* \mathbf{V}_{k+1}^* \mathbf{V}_{k+1} Z = Z^* (I - \mathbf{V}_{k+1}^* \mathbf{V}_{k+1}) Z + (I - Z^* Z) ,$$

which shows the theorem.  $\square$

When  $Z$  is square, Theorem 4.5 always holds. When, in addition to the conditions of Theorem 4.5,  $Z^* Z = I$ ,  $\mathcal{T}_Z \mathbf{Q}_k(S) \subset \mathbf{Q}_p(S)$ .

**4.1.3. Modification of the vectors of a Q-Arnoldi triple.** Let  $\{V_{k+1}, \underline{H}_k, w_{k+1}\} \in \mathbf{Q}_k(S)$ . Suppose we modify the second block of the Arnoldi vectors as follows:  $\tilde{W}_k = W_k + v_{k+1} g^*$ . Not surprisingly, (4.1), (3.1) and (4.2) are broken. With  $\tilde{\mathbf{V}}_k = \begin{pmatrix} V_k \\ \tilde{W}_k \end{pmatrix}$ , we have that the recurrence relation becomes

$$S \tilde{\mathbf{V}}_k - \tilde{\mathbf{V}}_{k+1} \underline{H}_k = G_k := \begin{pmatrix} v_{k+1} g^* \\ S_2 v_{k+1} g^* - v_{k+1} g^* H_k \end{pmatrix}$$

where

$$\|G_k\| \leq (\|S_2 v_{k+1}\| + \|v_{k+1}\| \|H_k\| + 1) \|g\| . \quad (4.5)$$

The structure of the vectors (4.1) can be restored by modifying  $\tilde{\underline{H}}_k = \underline{H}_k + e_{k+1}g^*$ . The recurrence relation for the new  $\underline{H}_k$  becomes

$$S\tilde{\mathbf{V}}_k - \tilde{\mathbf{V}}_{k+1}\tilde{\underline{H}}_k = \tilde{G}_k := \begin{pmatrix} 0 \\ S_2v_{k+1}g^* - v_{k+1}g^*H_k - w_{k+1}g^* \end{pmatrix}. \quad (4.6)$$

The upper bound (4.5) also holds for  $\tilde{G}_k$ . The orthogonality of  $\tilde{\mathbf{V}}_k$  is restored by applying  $\mathcal{T}_{Z^{-1}}$  to the Q-Arnoldi triple where  $Z$  is upper triangular and so that  $\tilde{\mathbf{V}}_k^*\tilde{\mathbf{V}}_k = Z^*Z$ , which can be computed by a Cholesky factorization.

Suppose we modify the first block of the Arnoldi vectors as follows:  $\hat{\mathbf{V}}_k = \mathbf{V}_k + v_{k+1}g^*$ , then (4.1) is restored by modifying  $\hat{\underline{H}}_k = \underline{H}_k - e_{k+1}g^*H_k$ . With  $\hat{\mathbf{V}}_k = \begin{pmatrix} \tilde{\mathbf{V}}_k \\ W_k \end{pmatrix}$ , we have that the recurrence relation for the new vectors becomes

$$S\hat{\mathbf{V}}_k - \hat{\mathbf{V}}_{k+1}\hat{\underline{H}}_k = \begin{pmatrix} 0 \\ S_1v_{k+1}g^* - w_{k+1}g^*H_k \end{pmatrix}. \quad (4.7)$$

The orthogonality can be restored in the same way as for  $\tilde{\mathbf{V}}_k$ .

## 4.2. Computing Ritz vectors and Schur vectors.

**4.2.1. Ritz vectors.** The Ritz vectors corresponding to Ritz value  $\theta$  have the form  $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} V_k z \\ V_{k+1}\underline{H}_k z \end{pmatrix}$  where  $H_k z = \theta z$ . When the Ritz value is an eigenvalue,  $x_2 = \theta x_1$ . As a Ritz vector of (1.1), we can return  $x_2/\theta$  or  $x_1$ .

In this section, we study which of these is the best choice. Let the residual of the Ritz pair computed by the Arnoldi method be

$$\begin{pmatrix} r_1 \\ r_2 \end{pmatrix} = \begin{bmatrix} 0 & I \\ S_1 & S_2 \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \theta \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_2 - \theta x_1 \\ S_1 x_1 + S_2 x_2 - \theta x_2 \end{pmatrix}. \quad (4.8)$$

If we use  $x_1$  as a Ritz vector for (1.1), the Ritz vector for the linearization is then

$$\tilde{x} = \begin{pmatrix} x_1 \\ \theta x_1 \end{pmatrix}.$$

The residual is

$$\begin{bmatrix} 0 & I \\ S_1 & S_2 \end{bmatrix} \begin{pmatrix} x_1 \\ \theta x_1 \end{pmatrix} - \theta \begin{pmatrix} x_1 \\ \theta x_1 \end{pmatrix} = \begin{pmatrix} 0 \\ -(S_2 - \theta I)r_1 + r_2 \end{pmatrix}$$

If we use  $x_2/\theta$  as a Ritz vector for (1.1), the Ritz vector for the linearization is then

$$\hat{x} = \begin{pmatrix} \theta^{-1}x_2 \\ x_2 \end{pmatrix}.$$

The residual is

$$\begin{bmatrix} 0 & I \\ S_1 & S_2 \end{bmatrix} \begin{pmatrix} \theta^{-1}x_2 \\ x_2 \end{pmatrix} - \theta \begin{pmatrix} \theta^{-1}x_2 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ \theta^{-1}S_1 r_1 + r_2 \end{pmatrix}$$

The conclusion from this analysis is that for large  $\theta$  there may be an advantage in using  $x_2/\theta$  and for small  $\theta$  in using  $x_1$ . Note that if  $x$  is close enough to an eigenvector,

then  $\|\tilde{x}\| \approx \|\hat{x}\|$ . Alternatively, the norms  $\|S_1\|$  and  $\|S_2\|$  may also play a role in the decision to make the choice.

Related to the choice of the first or second component is the difference  $x_2 - \theta x_1$ . If this difference is small, it probably does not make much difference which component we take.

**THEOREM 4.6.** *Let  $x_1 = V_k z$  and  $x_2 = W_k z$  with  $H_k z = \theta z$  and  $\|z\| = 1$  and define  $\rho = \beta_k |e_k^* z|$ .*

$$\|x_2 - \theta x_1\|_1 \leq \xi_{k,\min}^{-1} \rho$$

*Proof.* From (4.8), we have that

$$\begin{aligned} \|x_2 - \theta x_1\|_2 &\leq \|r_1\|_2 \\ &\leq \beta_k \|v_{k+1}\|_2 e_k^* z . \end{aligned}$$

□

We conclude that  $\|x_2 - \theta x_1\|_2$  is at most  $\rho$ , but can be smaller. Recall that

$$\xi_{k,\min} = \sqrt{1 + \sigma_{\min}^2(\underline{H}_k)} > 1 ,$$

which is large when the singular values of  $\underline{H}_k$  are large.

**4.2.2. Schur decomposition.** The eigenvectors usually do not form an orthogonal set of vectors and are not even guaranteed to exist (in the defective case). The Schur decomposition always produces orthogonal vectors and always exists.

For the linearized quadratic eigenvalue problem, the Schur decomposition of  $S$  is

$$S \begin{pmatrix} U \\ UT \end{pmatrix} = \begin{pmatrix} U \\ UT \end{pmatrix} T .$$

The diagonal elements of  $T$  are the Ritz values. (In the case of real matrices,  $T$  is in pseudo-upper triangular form when a Ritz value is complex. For the details see [5].)

Let  $H_k Z_k = Z_k T_k$  be the Schur decomposition of  $H_k$ . Define the residual  $r_k^* = \beta_k e_k^* Z_k$ . The Schur vectors computed from a Q-Arnoldi triple have the form

$$\mathbf{U}_k = \mathbf{V}_k Z_k = \begin{pmatrix} V_k Z_k \\ V_k Z_k T_k + v_{k+1} r_k^* \end{pmatrix} . \quad (4.9)$$

The structure in the Schur vectors is lost in the Krylov subspace for the same reasons as the Ritz vectors. However, we can similarly select the upper or lower components as Schur vectors. When we do this, we do not only add an error in the recurrence relation but also perturb the orthogonality of the basis vectors. We study this problem in detail in §4.4.

### 4.3. Implicit restarting.

**4.3.1. Implicit QR step.** When  $k$  gets large, the storage and computational costs of the Arnoldi method can become unacceptably high. Therefore, some form of restarting or reduction of the basis is desirable. The idea is to reduce the Krylov subspace by throwing away that part that is very unlikely to significantly contribute to the convergence of the wanted eigenvalues.



One way to perform such a reduction is called implicit restarting and was proposed by Sorensen [21]. Also see [21] [18] [10] [11] [16]. The idea is to apply an orthogonal transformation to  $H_k$  that pushes the  $p$  desired Ritz values of  $H_k$  to the principle  $p \times p$  block. The orthogonal transformation produces a reduction of the subspace of dimension  $k + 1$  to  $p + 1$  keeping the  $p$  desired Ritz values.

When  $Z$  results from the QR factorization  $\underline{H}_k - \mu \underline{I}$ , we have that

$$\underline{Z}_k^* \underline{H}_k = R_k + \mu \underline{Z}_k^* \underline{I},$$

and so

$$\underline{Z}_k \underline{Z}_k^* \underline{H}_k \underline{Z}_k \underline{Z}_k^{-1} = \underline{Z}_k (R_k \underline{Z}_k^{-1} + \mu I) = \underline{H}_k \underline{Z}_k^{-1}.$$

So, if  $\mathcal{Q} = \{V_{k+1}, \underline{H}_k, w_{k+1}\} \in \mathbf{Q}_k(s)$ , following Theorem 4.5,  $\mathcal{T}_Z \mathcal{Q} \in \mathbf{Q}_{p=k-1}(S)$ .

**4.3.2. Purging.** Another way to reduce the subspace dimension is purging [10]. The idea here is to purge the undesired part of the Schur factorization of  $H_k$ .

Recall the definitions of  $T_k$ ,  $Z_k$  and  $r_k$  from §4.2.2. By multiplying on the right by  $Z_k$ , the Arnoldi recurrence relation (3.1) can be written in terms of Schur vectors as follows :

$$S \mathbf{V}_k Z_k - \mathbf{V}_k Z_k T_k = \mathbf{v}_{k+1} r_k^*. \quad (4.10)$$

Let the Schur form be ordered so that the last  $k - p$  diagonal elements in  $T_k$  are unwanted Ritz values. The idea of purging is to keep the first  $p$  Schur vectors in the basis. Removing the last  $k - p$  columns from (4.10) produces

$$S \mathbf{V}_k Z_p - [\mathbf{V}_k Z_p \quad \mathbf{v}_{k+1}] \begin{pmatrix} T_p \\ r_p^* \end{pmatrix} = 0. \quad (4.11)$$

where  $Z_p$  are the first  $p$  columns of  $Z$ ,  $T_p$  is the leading  $p \times p$  block of  $T_k$  and  $r_p$  are the first  $p$  elements of  $r_k$ . There exists a unitary  $U_p$  so that

$$\tilde{\underline{H}}_p = \begin{pmatrix} U_p & 0 \\ 0 & 1 \end{pmatrix}^* \begin{pmatrix} T_p \\ r_p^* \end{pmatrix} U_p$$

is  $p + 1 \times p$  upper Hessenberg [5].

Let  $\mathcal{T}$  have transformation matrix  $\begin{pmatrix} Z_p U_p & 0 \\ 0 & 1 \end{pmatrix}$  (Definition 4.4). From

$$\underline{H}_k Z_p = \begin{bmatrix} Z_p & 0 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} T_p \\ r_p^* \end{pmatrix}$$

we derive that

$$\begin{bmatrix} Z_p & \\ & 1 \end{bmatrix} \begin{bmatrix} Z_p & \\ & 1 \end{bmatrix}^* \underline{H}_k Z_p = \underline{H}_k Z_p$$

and so

$$\begin{bmatrix} Z_p U_p & \\ & 1 \end{bmatrix} \begin{bmatrix} Z_p U_p & \\ & 1 \end{bmatrix}^* \underline{H}_k Z_p U_p = \underline{H}_k Z_p U_p.$$

So,  $\mathcal{T}\{V_{k+1}, \underline{H}_k, w_{k+1}\} \in \mathbf{Q}_p(S)$ .

**4.4. Locking.** Suppose that the first  $l$  elements in  $r_k$  are smaller than a convergence tolerance. The idea of locking is to set these small elements explicitly to zero assuming that the Schur vectors are exact. Decompose  $r_k^* = [r_l^* \ r_{k-l}^*]$ . Locking introduces an error in the recurrence relation :

$$S\mathbf{U}_k - [\mathbf{U}_k \ \mathbf{v}_{k+1}] \begin{pmatrix} T_k \\ [0 \ r_{k-l}^*] \end{pmatrix} = \mathbf{v}_{k+1}(r_l^* \ 0). \quad (4.12)$$

This can be considered as a backward error on the solution. The corresponding Q-Arnoldi triple is in  $\mathbf{Q}(S, \|r_l\|, 0)$ . The reasons for the use of locking in eigenvalue codes are the reduction of the dimension of the Krylov subspace and the computation of multiple eigenvalues without the need of block methods. Also, in combination with restarting, locking ensures that the residual norm of a Ritz pair of interest remains small : it is possible that with restarting a direction that is in favour of improving a Ritz vector is removed.

For linear problems it is accepted that the recurrence relation of Arnoldi's method is violated by a small term. The residual term in the right-hand side of (4.12) is usually considered as a backward error on  $S$ . The Bauer-Fike Theorem [20, Theorem 3.6] shows an upper bound to the perturbation of the eigenvalues that is proportional to  $\|r_l\|$  and that is small when the eigenvectors associated with different eigenvalues are almost orthogonal. The Schur vectors form an orthogonal set, so the orthogonality is preserved. In the following, we aim to preserve the structure of the Schur vectors (4.9). As we will see, this modifies the error on the recurrence relation, and destroys the orthogonality. The goal of the section is to restore the orthogonality and analyse the impact on the recurrence relation using the results from §4.1.3.

When we use the upper component as Schur vectors of the quadratic eigenvalue problem, the Schur vectors obtain the form

$$\tilde{\mathbf{U}}_k = \begin{pmatrix} V_k Z_l & U_{1,k-l} \\ V_k Z_l T_l & U_{2,k-l} \end{pmatrix} = \mathbf{U}_k - \begin{pmatrix} 0 & 0 \\ v_{k+1} r_l^* & 0 \end{pmatrix}. \quad (4.13)$$

Following (4.6) with  $g^* = -[r_l^* \ 0]$ , we find

$$\tilde{\mathbf{H}}_k = \begin{pmatrix} T_k \\ [0 \ r_{k-l}^*] \end{pmatrix}.$$

This is precisely the matrix we want to have with locking. Recall that  $\|\tilde{G}_k\| \simeq \|G_k\| \sim \|r_l\|$  is small. The new basis can be orthogonalized by applying an appropriate transformation.

Similarly, we can use the lower part of  $\mathbf{U}_k$  as Schur vectors. Decompose

$$Z_k = [Z_l \ Z_{k-l}] \quad \text{and} \quad T_k = \begin{bmatrix} T_l & T_{l,k-l} \\ & T_{k-l} \end{bmatrix}.$$

Now define Schur vectors using the lower part of  $\mathbf{U}_l$  :

$$\hat{\mathbf{U}}_k = \begin{pmatrix} V_k Z_l + v_{k+1} r_l^* T_l^{-1} & U_{1,k-l} \\ V_k Z_l T_l + v_{k+1} r_l^* & U_{2,k-l} \end{pmatrix} = \mathbf{U}_k - \begin{pmatrix} v_{k+1} r_l^* T_l^{-1} & 0 \\ 0 & 0 \end{pmatrix}. \quad (4.14)$$

This assumes that  $T_l$  is non-singular. With  $g^* = -[r_l^* T_l^{-1} \ 0]$ , following (4.7), the structure of the vectors is restored by using

$$\hat{\mathbf{H}}_k = \begin{pmatrix} T_k \\ [0 \ \hat{r}_{k-l}^*] \end{pmatrix},$$

$\tau$	1		$10^4$		$10^6$	
$\kappa(\underline{H})$	30.0		$1 \cdot 10^8$		$1 \cdot 10^{12}$	
$\ \underline{H}\ _2$	1.6		$1 \cdot 10^8$		$1 \cdot 10^{12}$	
$\frac{\sqrt{\gamma_m^2 + \delta_m^2 + 1.0}}{\sqrt{m+1}}$	1.03		$3.66 \cdot 10^6$		$1.66 \cdot 10^{10}$	
	$\gamma_\perp$	$\gamma_R$	$\gamma_\perp$	$\gamma_R$	$\gamma_\perp$	$\gamma_R$
Arnoldi	$1 \cdot 10^{-13}$	$1 \cdot 10^{-16}$	$5 \cdot 10^{-14}$	$1 \cdot 10^{-16}$	$2 \cdot 10^{-13}$	$2 \cdot 10^{-16}$
Q-Arnoldi	$3 \cdot 10^{-13}$	$1 \cdot 10^{-16}$	$4 \cdot 10^{-10}$	$6 \cdot 10^{-12}$	3.0	$4 \cdot 10^{-6}$

TABLE 5.1

Illustration of instabilities in the Q-Arnoldi method.  $\gamma_\perp = \|I - \mathbf{V}_{k+1}^* \mathbf{V}_{k+1}\|_F$  is the deviation of orthogonality and  $\gamma_R$  the error on the recurrence relation.

with  $\hat{r}_{k-l}^* = r_{k-l}^* - r_l^* T_l^{-1} T_{l,k-l}$ . The most important point is that the residual term corresponding to the first  $l$  Ritz values is set equal to zero. The residual terms of the remaining Ritz values are modified. This is not the case when  $\tilde{\mathbf{U}}_k$  is used as Schur basis. The error on the recurrence relation is again proportional to  $\|r_l\|$ .

## 5. Numerical examples.

**5.1. Illustration of numerical instabilities in Q-Arnoldi.** From §3.4, we can see that the Q-Arnoldi method may lose stability when  $\gamma_k$  is large. This is only possible when  $\|\underline{H}\|_2$  and  $\kappa(\underline{H}_k)$  are large.

We have run  $k = 10$  iterations of the Arnoldi method for a problem of dimension  $n = 10,000$  with  $K = I$ ,  $C = \tau I$  and  $M = \tau \text{diag}(\mu_1, \dots, \mu_n)$  where  $\mu_j = 1/j$  for three values of  $\tau$ . Table 5.1 shows (3.6). The (traditional) Arnoldi algorithm always produces accurate results. The Q-Arnoldi algorithm is sensitive to large  $\tau$ 's. The example illustrates that scaling the matrices may help improve the numerical stability of the Q-Arnoldi algorithm : indeed, the eigenvectors are the same for all cases, independent of  $\tau$ , and the eigenvalues of (1.1) are divided by  $\tau$ , but when  $\tau = 1$ ,  $K$ ,  $C$ , and  $M$  have norms around one.

**5.2. Selection of component of Ritz vectors.** Consider the quadratic eigenvalue problem (1.1) with  $K = I$ ,  $M = \text{diag}(\mu_1, \dots, \mu_n)$ , and  $C = 0.01M$  with  $\mu_j = 1/j$  for  $n = 10,000$ . We have run 10 steps of the Arnoldi method with an initial vector with equal components. Let  $(\theta, x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix})$  be a Ritz vector returned by the Arnoldi method. Define

$$\begin{aligned} \rho &= \|A^{-1}(\theta A - B)x\| \\ \rho_1 &= \|K^{-1}(\theta^2 K + \theta C + M)x_1\| \\ \rho_2 &= \|K^{-1}(\theta^2 K + \theta C + M)x_2\|. \end{aligned}$$

Table 5.2 shows the Ritz values and the corresponding residual norms. For this example,  $x_2$  shows to be a better Ritz vector than  $x_1$  for the large Ritz values.

**5.3. Quadratic eigenvalue problem for an acoustic box.** In this section we study the problem of an acoustic box with walls covered with carpet with dimensions  $0.54\text{m} \times 0.54\text{m} \times 0.55\text{m}$ . The material has a complex speed of sound  $340 + i3.4$  and the density is  $1.225\text{kg/m}^3$ . The box is discretized with 64710 hexahedral elements.

The matrices are produced by ACTRAN [4]. The problem to be solved has the form (1.1) and  $n = 13,623$ . We want to compute the 50 eigenvalues with imaginary

Ritz value	$\rho$	$\rho_2$	$\rho_1$
1.0099	$2 \cdot 10^{-7}$	$3 \cdot 10^{-8}$	$2 \cdot 10^{-7}$
0.509808	$1 \cdot 10^{-4}$	$2 \cdot 10^{-5}$	$7 \cdot 10^{-5}$
0.342946	$3 \cdot 10^{-3}$	$6 \cdot 10^{-4}$	$1 \cdot 10^{-3}$
0.256429	$2 \cdot 10^{-2}$	$3 \cdot 10^{-3}$	$5 \cdot 10^{-3}$
0.183618	$3 \cdot 10^{-2}$	$5 \cdot 10^{-3}$	$7 \cdot 10^{-3}$
0.117528	$3 \cdot 10^{-2}$	$5 \cdot 10^{-3}$	$4 \cdot 10^{-3}$
-0.00203237	$7 \cdot 10^{-5}$	$5 \cdot 10^{-4}$	$1 \cdot 10^{-5}$
0.00250052	$5 \cdot 10^{-3}$	$6 \cdot 10^{-5}$	$2 \cdot 10^{-4}$
$0.0381711 \pm i0.00132763$	$3 \cdot 10^{-2}$	$2 \cdot 10^{-3}$	$2 \cdot 10^{-3}$

TABLE 5.2  
Comparison of Ritz vectors

TABLE 5.3  
 $\sqrt{\gamma_k^2 + \delta_k^2}$  for the different restarts in the solution of the quadratic eigenvalue problem of the box

before restart	2.14777
first restart	1.93901
second restart	1.73840
third restart	1.69189

part larger than 600. We applied the shift  $\sigma = 600i$ , see §3.3. We used the following algorithm.

ALGORITHM 5.1 (Arnoldi for eigenvalues).

1. Choose  $\mathbf{v}_1$  randomly and normalize.
2. Until the wanted eigenvalues have converged, do :
  - 2.1. Build a Krylov subspace of dimension  $k$ .
  - 2.1. Compute Ritz values, Ritz vectors and residual norms.
  - 2.2. Order the Ritz values in increasing distance to  $\sigma$ .
  - 2.3. Purge the last  $m - p$  columns of the recurrence relation.

We solved the problem using the Arnoldi method with  $k = 100$  and  $p = 50$ . The first iteration costs  $k = 100$  products with  $S$ . In Step 2.3, the purging operation keeps  $p$  iteration vectors with Ritz values corresponding to the Ritz values nearest  $\sigma$ . The goal of the next iterations is to improve these values. The next call to Step 2.1 requires only  $k - p$  additional iterations to obtain a subspace of dimension  $k$ . After 3 restarts, 50 Ritz values were computed with residual norms smaller than  $10^{-8}$ . The computations were carried out on a Linux PC. The final loss of orthogonality in the Q-Arnoldi algorithm was

$$\|I - \mathbf{V}_{k+1}^* \mathbf{V}_{k+1}\|_F \simeq 3.1 \cdot 10^{-13} .$$

For the Arnoldi algorithm we also had

$$\|I - \mathbf{V}_{k+1}^* \mathbf{V}_{k+1}\|_F \simeq 3.1 \cdot 10^{-13} .$$

Table 5.3 shows  $\sqrt{\gamma_k^2 + \delta_k^2}$  for the different restarts. For all restarts,  $\sqrt{\gamma_k^2 + \delta_k^2}$  is small, so we do not expect numerical difficulties. This is no surprise since the shift  $\sigma = 600i$  is not close to an eigenvalue of (1.1), so  $\|\underline{H}_k\|_2 \leq \|S\|_2$  is small.

**6. Conclusions.** The Q-Arnoldi algorithm is a memory efficient implementation of the Arnoldi method for specific choices of linearization of the quadratic eigenvalue problem.

We have proposed an algorithm that preserves the structure of the Schur vectors, and that shows that implicit restarting, purging and locking similarly preserve the structure of the Arnoldi vectors.

As for the choice of linearization, due to  $A^{-1}$ , the Arnoldi method produces the same results for any linearization in  $\mathbb{L}_1$ . The same conclusion holds for the SOAR method. This also implies (as we already knew) that the Arnoldi method in its standard form is not able to preserve structure.

An important conclusion lies in the influence of  $\xi_{\min}$  and  $\xi_{\max}$ . The ratio should not be far away from one in order to reduce the chance of cancelation in the numerical computations. In addition, the components of the Ritz vectors lie in the same direction when  $\xi_{\min}$  is large. However, as we mentioned earlier, for the shift-and-invert transformation,  $\xi_{\min}$  usually lies close to one. We have some freedom in choosing the pole  $\sigma$  to keep  $\xi_{\max}$  low. The derivation of scalings of  $S_1$  and  $S_2$  is still an open problem. Note that  $S_1$  and  $S_2$  are only known in factored form  $-K^{-1}C$  and  $-K^{-1}M$  respectively.

The conclusion is not the SOAR method is useless when more than one eigenvalue needs to be computed or restarting the Arnoldi process is required. The Q-Arnoldi algorithm produces a Krylov subspace, whereas the SOAR method projects  $K$ ,  $C$  and  $M$  on  $V_{k+1}$ . This is still possible in a post-processing step in the Q-Arnoldi algorithm in order to improve the Ritz values or impose spectral structure.

Open problems are restarting the SOAR method, and other linearizations than those in  $\mathbb{L}_1$ , e.g.

$$\left\{ I - \lambda S : (I - \lambda S) \begin{pmatrix} \alpha_0 + \lambda\alpha_1 \\ \beta_0 + \lambda\beta_1 \end{pmatrix} = \begin{pmatrix} \eta_1 Q(\lambda) \\ \eta_2 Q(\lambda) \end{pmatrix}, \alpha_{0,1}, \beta_{0,1}, \eta_{1,2} \in \mathbb{C} \right\}$$

The extension to higher order polynomials,

$$(A_0 + \lambda A_1 + \dots + \lambda^p)u = 0$$

is straightforward and leads to a similar algorithm with similar conclusions.

**Acknowledgement.** This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its author(s).

The author is also grateful to Mickaël Robbé who helped with an early draft of the paper.

REFERENCES

[1] W. ARNOLDI, *The principle of minimized iterations in the solution of the matrix eigenvalue problem*, Quart. Appl. Math., 9 (1951), pp. 17–29.  
 [2] Z. BAI, J. DEMMEL, J. DONGARRA, A. RUHE, AND H. VAN DER VORST, *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, SIAM, Philadelphia, PA, USA, 2000.  
 [3] Z. BAI AND Y. SU, *A second-order Arnoldi method for the solution of the quadratic eigenvalue problem*, SIAM J. Matrix Anal. Applic., 26 (2005), pp. 640–659.  
 [4] FREE FIELD TECHNOLOGIES, *MSC.Actran 2006, User’s Manual*, 2006.

- [5] G. GOLUB AND C. VAN LOAN, *Matrix computations*, The Johns Hopkins University Press, 3rd ed., 1996.
- [6] R. GRIMES, J. LEWIS, AND H. SIMON, *A shifted block Lanczos algorithm for solving sparse symmetric generalized eigenproblems*, SIAM J. Matrix Anal. Applic., 15 (1994), pp. 228–272.
- [7] N. HIGHAM, *Accuracy and Stability of Numerical Algorithms*, SIAM, Philadelphia, PA, USA, 1996.
- [8] C. LANCZOS, *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Res. Nat. Bur. Stand., 45 (1950), pp. 255–282.
- [9] ———, *Solution of systems of linear equations by minimized iterations*, J. Res. Nat. Bur. of Stand., 49 (1952), pp. 33–53.
- [10] R. LEHOUCQ AND D. SORENSEN, *Deflation techniques within an implicitly restarted Arnoldi iteration*, SIAM J. Matrix Anal. Applic., 17 (1996), pp. 789–821.
- [11] R. B. LEHOUCQ, D. C. SORENSEN, AND C. YANG, *ARPACK USERS GUIDE: Solution of Large Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*, SIAM, Philadelphia, PA, USA, 1998.
- [12] D. S. MACKEY, N. MACKEY, C. MEHL, AND V. MEHRMANN, *Structured polynomial eigenvalue problems: Good vibrations from good linearizations*, SIAM J. Matrix Anal. Applic., 28 (2006), pp. 1029–1051.
- [13] ———, *Vector spaces of linearizations for matrix polynomials*, SIAM J. Matrix Anal. Applic., 28 (2006), pp. 971–1004.
- [14] K. MEERBERGEN, *Locking and restarting quadratic eigenvalue solvers*, SIAM J. Sci. Comput., 22 (2001), pp. 1814–1839.
- [15] ———, *The rational Lanczos method for Hermitian eigenvalue problems*, Num. Lin. Alg. Appl., 8 (2001), pp. 33–52.
- [16] K. MEERBERGEN AND A. SPENCE, *Implicitly restarted Arnoldi and purification for the shift-invert transformation*, Math. Comp., 66 (1997), pp. 667–689.
- [17] V. MEHRMANN AND D. WATKINS, *Structure-preserving methods for computing eigenpairs of large sparse skew-Hamiltonian/Hamiltonian pencils*, SIAM J. Matrix Anal. Applic., 22 (2001), pp. 1905–1925.
- [18] R. MORGAN, *On restarting the Arnoldi method for large nonsymmetric eigenvalue problems*, Math. Comp., 65 (1996), pp. 1213–1230.
- [19] B. N. PARLETT AND H. C. CHEN, *Use of indefinite pencils for computing damped natural modes*, Linear Alg. Appl., 140 (1990), pp. 53–88.
- [20] Y. SAAD, *Numerical methods for large eigenvalue problems*, Algorithms and Architectures for Advanced Scientific Computing, Manchester University Press, Manchester, UK, 1992.
- [21] D. SORENSEN, *Implicit application of polynomial filters in a  $k$ -step Arnoldi method*, SIAM J. Matrix Anal. Applic., 13 (1992), pp. 357–385.
- [22] F. TISSEUR, *Backward error and condition of polynomial eigenvalue problems*, Linear Alg. Appl., 309 (2000), pp. 339–361.
- [23] F. TISSEUR AND K. MEERBERGEN, *The quadratic eigenvalue problem*, SIAM Rev., 43 (2001), pp. 235–286.
- [24] D. Z. BAI AND Q. YE, *ABLE: an adaptive block lanczos method for non-hermitian eigenvalue problems*, SIAM J. Matrix Anal. Applic., 20 (1999), pp. 1060–1082.