# RESTARTING AN ARNOLDI REDUCTION *

R. B. LEHOUCQ†

**Abstract.** The Arnoldi reduction is an efficient procedure for approximating a subset of the eigensystem of a large sparse matrix $\mathbf{A}$ of order $n$. At each step, a partial orthogonal reduction of $\mathbf{A}$ into an upper Hessenberg matrix is produced. The eigenvalues of this Hessenberg matrix are used to approximate a subset of the eigenvalues of the large matrix $A$. The approximation to the eigenvalues of $\mathbf{A}$ generally improves as the order of the Hessenberg matrix increases. Unfortunately, so do the cost and storage of the reduction.

A popular alternative is to define an iteration by restarting the reduction with information in a length $m < n$ Arnoldi reduction. The hope is that this restarted reduction has improved estimates to the eigenvalues of $\mathbf{A}$.

This paper considers the various approaches used to restart a reduction. Analysis and numerical examples are presented that explain and exhibit the generally superior properties of Sorensen's implicitly restarted Arnoldi iteration. The analysis exploits the fact that an IRA iteration is mathematically equivalent to a curtailed QR iteration.

**Key words.** Arnoldi reduction, Lanczos reduction, restarting, eigenvalues.

**AMS subject classifications.** 65F15, 65G05

**1. Introduction.** The Arnoldi reduction [2] is an orthogonal projection method for approximating a subset of the eigensystem of a general square matrix. Starting with a vector $\mathbf{x}_1$, the reduction builds, step by step, an orthogonal basis for the *Krylov* subspace of $\mathbf{A}$:

$$\mathcal{K}_m(\mathbf{A}, \mathbf{x}_1) \equiv \mathrm{Span}\{\mathbf{x}_1, \mathbf{A}\mathbf{x}_1, \ldots, \mathbf{A}^{m-1}\mathbf{x}_1\}.$$

It is a generalization of the power method in that a sequence of iterates are used to approximate eigenvalues of $\mathbf{A}$. At every step of the reduction, the projection of $\mathbf{A}$ onto $\mathcal{K}_m(\mathbf{A}, \mathbf{x}_1)$ is computed. This projection is an upper Hessenberg matrix of order $m$. The eigenvalues of this projection matrix are used as approximations to those of $\mathbf{A}$. Since the reduction requires knowledge of $\mathbf{A}$ only through matrix-vector products, its value as a technique for approximating a few eigenvalues of a large sparse matrix was soon realized. When the matrix $\mathbf{A}$ is symmetric, the Lanczos reduction [15] is recovered.

More than a decade of research has been devoted to understanding and overcoming the numerical difficulties of the Lanczos reduction. The works of Parlett [30] and Cullum and Wiloughby [8] study in detail the many specifics of the Lanczos algorithm, while the paper by Grimes, Lewis, and Simon [13] discusses the design and development of high-quality software.

Development of the Arnoldi reduction lagged behind because of the inordinate computational and storage requirements associated with the original method when a large number of steps are required for convergence. The explicitly restarted Arnoldi iteration (ERA iteration) was introduced by Saad [32] to overcome these difficulties, based on similar ideas developed for the Lanczos process by Paige [27], Cullum and Donath [7], and Golub and Underwood [11]. Karush [14] proposes what appears to

* This work was supported in part by ARPA (U.S. Army ORA4466.01), by the U.S. Department of Energy (Contracts DE-FG0f-91ER25103 and W-31-109-Eng-38), and by the National Science Foundation (Cooperative agreement CCR-9120008).

† Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439 lehoucq@mcs.anl.gov.

be the first example of a restarted iteration. The iteration is defined by a two-stage process. First, an Arnoldi/Lanczos reduction of length $m < n$ is computed. From information available in this reduction, another reduction is computed. This defines the iteration and is deemed successful if improved estimates to the eigenvalues of $\mathbf{A}$ appear in the subsequent reductions.

A relatively recent variant was developed by Sorensen [39] as a more efficient and numerically stable way to implement restarting. This technique, the implicitly restarted Arnoldi iteration (IRA iteration), may be viewed as a truncation of the standard implicitly shifted QR-iteration. This viewpoint provides an alternate approach to study restarted Arnoldi/Lanczos reductions in which the power of the QR algorithm is used. The immediate effect is the improvement of the numerical accuracy and convergence properties of the `ARPACK` [19] software package.

The paper is organized as follows. Some notation and the real Schur decomposition are introduced in §2. The eigenvalue problem and Arnoldi reduction along with more notation are the subject of §§3–4. The ERA and IRA iterations are examined in §§5–6, and an example comparing them is given §7. Three theorems that characterize an IRA iteration are presented in §8. The polynomial accelerations techniques are reviewed in §9 along with an improved ERA iteration that incorporates a deflation scheme. Some results of a numerical study are given in §10, and a theorem on the convergence of an IRA iteration is presented in §11. The paper is concluded in §12 with a discussion on some important practical issues.

**2. Notation and Some Fundamentals.** We shall now establish the basic notation to be used during the course of this study.

We employ Householder notational conventions. Capital and lower-case letters denote matrices and vectors, respectively, while lower-case Greek letters denote scalars. For clarity, upper-case matrices have a subscript to indicate the number of columns. The identity matrix of order $m$ is denoted by $\mathbf{I}_m$. The $j$th canonical basis vector is denoted by $\mathbf{e}_j$, the $j$th column of the identity matrix.

The transpose of a vector $\mathbf{x}$ is denoted by $\mathbf{x}^T$, and $\mathbf{x}^H$ denotes the complex conjugate of $\mathbf{x}^T$. The norms used are the Euclidean and Frobenius, denoted by $\|\cdot\|$ and $\|\cdot\|_F$, respectively. The range of a matrix $\mathbf{A}$ is denoted by $\mathcal{R}(\mathbf{A})$.

**2.1. Real Schur Decomposition.** Since we are especially concerned with algorithms that result in robust and efficient software, we focus on the following decomposition, which is a special case of the more general Schur decomposition. The special case allows us to compute strictly in real arithmetic. The proper resolution of complex conjugate pairs of eigenvalues comes from noting that if $\mathbf{A}(\mathbf{x}+i\mathbf{z}) = (\nu+i\mu)(\mathbf{x}+i\mathbf{z})$, where $\mathbf{x}$ and $\mathbf{z}$ are vectors in $\mathbf{R}^n$ with $\mu \neq 0$, then

$$\mathbf{A}\begin{bmatrix} \mathbf{x} & \mathbf{z} \end{bmatrix} = \begin{bmatrix} \mathbf{x} & \mathbf{z} \end{bmatrix}\begin{bmatrix} \nu & \mu \\ -\mu & \nu \end{bmatrix}. \tag{2.1}$$

The following decomposition proves central to the eigenvalue algorithms considered in this paper. This decomposition is computed by the practical QR algorithm in the LAPACK [1] software library.

THEOREM 2.1. *(Real Schur Decomposition) If* $\mathbf{A} \in \mathbf{R}^{n\times n}$, *there exists an orthog-*

*onal* $\mathbf{Z} \in \mathbf{R}^{n\times n}$ *such that*

$$\mathbf{Z}^T\mathbf{A}\mathbf{Z} = \begin{bmatrix} T_{11} & T_{12} & \cdots & T_{1m} \\ 0 & T_{22} & \cdots & T_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & T_{mm} \end{bmatrix} \equiv \mathbf{T}, \tag{2.2}$$

*where each* $T_{ii}$ *is a square block of order one or two. The blocks of order two contain the complex conjugate eigenvalues of* $\mathbf{A}$*. The matrix* $\mathbf{T}$ *is said to be in upper* quasi–triangular *matrix form.*

*Proof.* See [10, page 362]. □

Let $\mathbf{C}$ be a quasi-diagonal orthogonal matrix with two by two blocks allowed only where $\mathbf{T}$ has them. Then $(\mathbf{ZC})^T\mathbf{AZC} = \mathbf{C}^T\mathbf{RC}$ has diagonal blocks that are similar to those of $\mathbf{T}$. Thus, apart from the eigenvalues of multiplicity larger than one, the decomposition is essentially unique, given some ordering of the eigenvalues. Denote the leading principal matrix of $k$ blocks of $\mathbf{T}$ by $\mathbf{T}_k$ where no $T_{ii}$ is split. Let $\mathbf{Z}_k$ be the corresponding columns of $\mathbf{Z}$. Then $\mathbf{A}\mathbf{Z}_k = \mathbf{Z}_k\mathbf{T}_k$ is a *partial* real Schur decomposition of $\mathbf{A}$ of order $k$.

**3. The Eigenvalue Problem.** Let $\mathbf{A}$ be a real matrix of order $n$. We are interested in a specified set of $k \ll n$ solutions to the matrix eigenvalue problem

$$\mathbf{A}\mathbf{u} = \lambda\mathbf{u}. \tag{3.1}$$

The eigenvalues and eigenvectors of $\mathbf{A}$ are denoted by $\lambda_j$ and $\mathbf{u}_j$, respectively, for $j = 1, \ldots, n$. We shall refer to these $k$ eigenvalues as the *wanted* ones. The wanted eigenvalues of $\mathbf{A}$ requiring approximation typically are contained within some convex set of interest in the complex plane. Examples include those nearest the origin, and of largest real part. An important exception might be the *dominant* eigenvalues of $\mathbf{A}$, those largest in magnitude.

The Arnoldi methods studied in this paper attempt to compute a partial real Schur decomposition for $\mathbf{A}$ with the group of the wanted eigenvalues located on the diagonal blocks of $\mathbf{T}_k$. The methods considered require storage of $\mathcal{O}(kn)$. The full decomposition requires $\mathcal{O}(n^2)$ storage. We say an eigenvalue problem is large if the dense QR algorithm is prohibative, either in storage and/or efficiency.

A *quasi-diagonal* form for $\mathbf{A}$ exists if there is a nonsingular matrix $\mathbf{V}$ such that $\mathbf{AV} = \mathbf{VD}$, where $\mathbf{D}$ is a block diagonal matrix with each block of order one or two. The blocks of order two contain the complex conjugate pair of eigenvalues as in equation (2.1) with $\mu$ positive. The columns of $\mathbf{V}$ span the right eigenspace corresponding to diagonal values of $\mathbf{D}$. For the blocks of order two on the diagonal of $\mathbf{D}$, the corresponding complex eigenvector is stored in two consecutive columns of $\mathbf{V}$, the first holding the real part, and the second the imaginary part.

**4. The Arnoldi Reduction.** We first show how the Arnoldi reduction is extended from length $m$ to $m+1$ where $m < n$. Table 4.1 summarizes the procedure. If $m = 1$, then $\mathbf{X}_1 = \mathbf{x}_1$ represents the initial vector. $\mathbf{H}_m$ is an upper Hessenberg matrix of order $m$ while $\mathbf{X}_m$ contains $m$ vectors of length $n$ that are mutually orthogonal. The residual vector $\mathbf{f}_m$ is orthogonal to the columns of $\mathbf{X}_m$. In order to ensure that $\mathbf{f}_m$ is orthogonal to the column space of $\mathbf{X}_m$ in finite-precision arithmetic, some form of reorthogonalization is necessary at step 5. See Chapter 7 of [16].

TABLE 4.1
*Extending an Arnoldi Reduction*

- Let $\mathbf{A}\mathbf{X}_m = \mathbf{X}_m\mathbf{H}_m + \mathbf{f}_m\mathbf{e}_m^T$ be a length $m$ Arnoldi reduction.
  1. $\beta_{m+1} = \|\mathbf{f}_m\|$.
  2. If $\beta_{m+1} \neq 0$, then $\mathbf{x}_{m+1} = \mathbf{f}_m\beta_{m+1}^{-1}$;
     else set $\mathbf{x}_{m+1}$ to be an unit vector orthogonal to the column space of $\mathbf{X}_m$.
  3. $\mathbf{X}_{m+1} = \left[\begin{array}{cc} \mathbf{X}_m & \mathbf{x}_{m+1} \end{array}\right]$.
  4. $\mathbf{w} = \mathbf{A}\mathbf{x}_{m+1}$.
  5. $\mathbf{H}_{m+1} = \left[\begin{array}{cc} \mathbf{H}_m & \mathbf{X}_m^T\mathbf{w} \\ \beta_{m+1}\mathbf{e}_m^T & \mathbf{x}_{m+1}^T\mathbf{w} \end{array}\right]$.
  6. $\mathbf{f}_{m+1} = \mathbf{w} - \mathbf{X}_{m+1}\mathbf{H}_{m+1}\mathbf{e}_{m+1}$.

The matrix $\mathbf{H}_m = \mathbf{X}_m^T\mathbf{A}\mathbf{X}_m$ is the orthogonal projection of $\mathbf{A}$ onto the $\mathcal{R}(\mathbf{X}_m) \equiv \mathcal{K}_m(\mathbf{A}, \mathbf{x}_1)$. The Hessenberg matrix $\mathbf{H}_m$ is said to be unreduced if all of its subdiagonal elements $\beta_m$ are nonzero. Note that the first subdiagonal element is $\beta_2$. The following classical result explains that an Arnoldi reduction is completely specified by the starting vector.

THEOREM 4.1. *(Implicit Q) Let two length m Arnoldi reductions be given by*

$$\mathbf{A}\mathbf{X}_m = \mathbf{X}_m\mathbf{H}_m + \mathbf{f}_m\mathbf{e}_m^T,$$
$$\mathbf{A}\mathbf{V}_m = \mathbf{V}_m\mathbf{G}_m + \mathbf{r}_m\mathbf{e}_m^T,$$

*where $\mathbf{X}_m$ and $\mathbf{V}_m$ have orthonormal columns, and $\mathbf{H}_m$ and $\mathbf{G}_m$ are upper Hessenberg matrices with positive subdiagonal elements. If the first columns of $\mathbf{X}_m$ and $\mathbf{V}_m$ are equal, then $\mathbf{H}_m = \mathbf{G}_m$ , $\mathbf{X}_m = \mathbf{V}_m$ and $\mathbf{f}_m = \mathbf{r}_m$.*

*Proof.* See [10, page 367]. □

Let $(\mathbf{s_j}, \theta_j)$ for $j = 1, \ldots, m$ be the eigenpairs of $\mathbf{H}_m$ where all the $\mathbf{s}_j$ are of unit length. The quality of the approximation of $\theta_j$ to an eigenvalue of $\mathbf{A}$ is determined by observing that

$$\|\mathbf{A}\mathbf{X}_m\mathbf{s}_j - \mathbf{X}_m\mathbf{s}_j\theta_j\| = \|\mathbf{f}_m\|\,|\mathbf{e}_m^T\mathbf{s}_j|.$$

We call $\mathbf{z}_j \equiv \mathbf{X}_m\mathbf{s}_j$ a *Ritz* vector, $\theta_j$ a *Ritz* value, and $\|\mathbf{f}_m\|\,|\mathbf{e}_m^T\mathbf{s}_j|$ the *Ritz* estimate. Thus we may easily compute the norm of the residual of a Ritz pair and avoid the application of $\mathbf{A}$. We remark that a tiny Ritz estimate does not imply an accurate approximation. We refer the reader to the work of Scott [37] for the many issues that must be considered for assessing the quality of Ritz pairs computed by the Arnoldi reduction.

As long as $\mathbf{H}_m$ is unreduced, standard results on upper Hessenberg matrices give that $\mathbf{e}_m^T\mathbf{s}_j$ cannot be equal to zero, although in practical computation, they decrease in size as $\theta_j$ better approximates an eigenvalue of $\mathbf{A}$. On the other hand, the vector $\mathbf{f}_m$ vanishes at the first step $m$ such that $\mathcal{K}_{m+1}(\mathbf{A}, \mathbf{x}_1) = \mathcal{K}_m(\mathbf{A}, \mathbf{x}_1)$ and hence is guaranteed to vanish for some $m \leq n$. The following result indicates when we may expect $\mathbf{f}_m = \mathbf{0}$. This is desirable because then the columns of $\mathbf{X}_m$ form an orthogonal basis for an invariant subspace and the eigenvalues of $\mathbf{H}_m$ are those of $\mathbf{A}$.

THEOREM 4.2. *Let $\mathbf{A}\mathbf{X}_m = \mathbf{X}_m\mathbf{H}_m + \mathbf{f}_m\mathbf{e}_m^T$ define a m-step Arnoldi reduction of $\mathbf{A}$, with $\mathbf{H}_m$ unreduced. Then $\mathbf{f}_m = \mathbf{0}$ if and only if $\mathbf{x}_1 \in \mathcal{R}(\mathbf{Z}_m)$, where $\mathbf{A}\mathbf{Z}_m = \mathbf{Z}_m\mathbf{T}_m$ is a partial real Schur decomposition.*

*Proof.* See Chapter 2 of [16] or [39] for a proof based on the Jordan canonical form. □

TABLE 5.1
*Explicitly Restarted Arnoldi Iteration*

- *Start:* Build a length $m$ Arnoldi reduction $\mathbf{A}\mathbf{X}_m = \mathbf{X}_m\mathbf{H}_m + \mathbf{f}_m\mathbf{e}_m^T$ with the starting vector $\mathbf{x}_1$.
- *Iteration:* Until convergence
  1. Compute the eigensystem $\mathbf{H}_m\mathbf{S}_m = \mathbf{S}_m\mathbf{D}_m$ ordered with the $k$ wanted eigenvalues located in the leading portion of the quasi-diagonal matrix $\mathbf{D}_m$.
  2. *Restart:* Select the new starting vector $\mathbf{x}_1 = \mathbf{X}_m\mathbf{y}_m$ where $\mathbf{y}_m \in \mathbf{R}^m$.
  3. Build a length $m$ Arnoldi reduction with $\mathbf{x}_1$.

In Theorem 4.2, the span of the $m$ columns of $\mathbf{Z}_m$ represents an invariant subspace for $\mathbf{A}$. In particular, the theorem gives that if the initial vector is a linear combination of $m$ eigenvectors, then $\mathbf{f}_m$ vanishes. It is therefore desirable to devise a method that forces the starting vector $\mathbf{x}_1$ to lie in the invariant subspace associated with the the wanted eigenvalues. We also remark that working in finite-precision arithmetic generally removes the possibility of the computed residual ever vanishing exactly. Suppose, however, that $\|\mathbf{f}_m\|$ is small. Let $\mathbf{H}_m\mathbf{V}_m = \mathbf{V}_m\mathbf{T}_m$ be a real Schur decomposition. Then $(\mathbf{A} + \mathbf{E}_m)\mathbf{X}_m\mathbf{V}_m = \mathbf{X}_m\mathbf{V}_m\mathbf{T}_m$ is an exact real partial Schur form for a nearby problem where $\mathbf{E}_m = -\mathbf{f}_m\mathbf{e}_m^T\mathbf{V}_m^T\mathbf{X}_m^T$. Note that $\|\mathbf{E}_m\| = \|\mathbf{f}_m\|$.

The algorithms of this paper are appropriate when the order of $\mathbf{A}$ is so large that storage and computational requirements prohibit completion of the algorithm that produces $\mathbf{X}_n$ and $\mathbf{H}_n$.

**5. An ERA Iteration.** The basic explicitly restarted Arnoldi iteration is summarised in Table 5.1.

The choice of $m$ is usually a tradeoff between the length of the reduction that may be tolerated and the rate of convergence. From the results on the convergence of Krylov spaces [33, 36], the accuracy of the Ritz values typically increases as $m$ does. However, for increasing $m$, the number of Arnoldi vectors stored as well as the size of the Hessenberg matrix increases. For most problems, the size of $m$ is determined experimentally. This issue is further addressed in §12.

The selection of the expansion coefficients in the vector $\mathbf{y}_m$ is the most unsettling decision that needs to be made. Saad first suggested [32] choosing the coefficients so that the slowest converging Ritz vectors are favored the most. For example, let $\mathbf{y}_m \leftarrow \mathbf{s}_1\gamma_1 + \cdots + \mathbf{s}_k\gamma_k$, where $\gamma_i = |\mathbf{e}_m^T\mathbf{s}_i|\,\|\mathbf{f}_m\| (= \|\mathbf{A}\mathbf{X}_m\mathbf{s}_i - \mathbf{X}_m\mathbf{s}_i\theta_i\|)$. The resulting vector $\mathbf{X}_m\mathbf{y}_m$ is a linear combination of the wanted Ritz vector. When $\theta_i$ has a nonzero imaginary part, we set $\mathbf{s}_i$ and $\mathbf{s}_{i+1}$ to be the real and imaginary portions of the complex eigenvector of $\mathbf{H}_m$ associated with $\theta_i$.

**6. An IRA Iteration.** The `ARPACK` software package [19] implements an implicitly restarted Arnoldi method. Table 6.1 gives the basic algorithm as implemented by `ARPACK`. The scheme is called *implicit* because the starting vector is updated with an implicitly shifted QR algorithm on the Hessenberg matrix $\mathbf{H}_m$. The method is motivated by the following result.

LEMMA 6.1. *Let $\mathbf{A}\mathbf{X}_m = \mathbf{X}_m\mathbf{H}_m + \mathbf{f}_m\mathbf{e}_m^T$ be a length $m$ Arnoldi reduction and $\psi(\cdot)$ a polynomial of degree $p = m - k$ where $k < m$.*

*If the* QR *factorization of $\psi(\mathbf{H}_m)\begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \cdots & \mathbf{e}_k \end{bmatrix} = \mathbf{Q}_k\mathbf{R}_k$, then the columns of $\mathbf{X}_m\mathbf{Q}_k$ are an orthogonal basis for $\mathcal{R}(\psi(\mathbf{A})\mathbf{X}_k)$.*

*Proof.* A straightforward induction argument shows that

$$\psi(\mathbf{A})\mathbf{X}_k = \mathbf{X}_m\psi(\mathbf{H}_m)\begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \cdots & \mathbf{e}_k \end{bmatrix}. \tag{6.1}$$

TABLE 6.1
*An Implicitly Restarted Arnoldi Iteration as Implemented by* ARPACK.

- *Start:* Build a length $m$ Arnoldi reduction $\mathbf{A}\mathbf{X}_m = \mathbf{X}_m\mathbf{H}_m + \mathbf{f}_m\mathbf{e}_m^T$ with the starting vector $\mathbf{x}_1$.
- *Iteration:* Until convergence
  1. Compute the eigensystem $\mathbf{H}_m\mathbf{S}_m = \mathbf{S}_m\mathbf{D}_m$ ordered with the $k$ wanted eigenvalues located in the leading portion of the quasi-diagonal matrix $\mathbf{D}_m$.
  2. Perform $m - k = p$ steps of the QR iteration with the unwanted eigenvalues of $\mathbf{D}_m$ as shifts to obtain $\mathbf{H}_m\mathbf{Q}_m = \mathbf{Q}_m\mathbf{H}_m^+$.
  3. *Restart:* Postmultiply the length $m$ Arnoldi reduction with $\mathbf{Q}_k$ to obtain the length $k$ Arnoldi reduction $\mathbf{A}\mathbf{X}_m\mathbf{Q}_k = \mathbf{X}_m\mathbf{Q}_k\mathbf{H}_k^+ + \mathbf{f}_k^+\mathbf{e}_k^T$. Note that $\mathbf{Q}_k$ represents the matrix consisting of the leading $k$ columns of $\mathbf{Q}_m$, and $\mathbf{H}_k^+$ is the leading principal submatrix of order $k$ of $\mathbf{H}_m^+$.
  4. Extend the length $k$ Arnoldi reduction to a length $m$ one.

Compute the QR factorization $\psi(\mathbf{H}_m)\left[\begin{array}{cccc}\mathbf{e}_1 & \mathbf{e}_2 & \cdots & \mathbf{e}_k\end{array}\right] = \mathbf{Q}_k\mathbf{R}_k$. Equation (6.1) may then be rewritten as $\psi(\mathbf{A})\mathbf{X}_k = \mathbf{X}_m\mathbf{Q}_k\mathbf{R}_k$, and the lemma is proved. □

This is a generalization of the special case $\psi(\lambda) = \lambda$ shown in [22]. A similar result was proved by Paige, Parlett, and Van der Vorst in Lemma 1 of [28] for the Lanczos reduction.

Restarting the iteration involves postmultiplying the length $m$ Arnoldi factorization with $\mathbf{Q}_k$ and thus obtaining a length $k$ factorization. The actual computation of $\mathbf{Q}_k$ involves performing $p$ steps of the QR algorithm on the current upper Hessenberg matrix, using the zeros of $\psi$ as shifts to obtain $\mathbf{H}_m\mathbf{Q}_m = \mathbf{Q}_m\mathbf{H}_m^+$. This allows us to exploit the well-known connection between the QR algorithm and subspace iteration [42]. Equate the first $k$ columns of $\mathbf{H}_m\mathbf{Q}_m = \mathbf{Q}_m\mathbf{H}_m^+$ to get $\mathbf{H}_m\mathbf{Q}_k = \mathbf{Q}_k\mathbf{H}_k^+ + \beta_{k+1}^+\mathbf{Q}_m\mathbf{e}_{k+1}\mathbf{e}_k^T$. Note that $\mathbf{e}_m^T\mathbf{Q}_k = (\mathbf{e}_m^T\mathbf{Q}_k\mathbf{e}_k)\mathbf{e}_k^T$, since $\mathbf{Q}_m$ is of lower bandwidth $k$, and thus

$$\begin{aligned}\mathbf{A}\mathbf{X}_m\mathbf{Q}_k &= \mathbf{X}_m\mathbf{H}_m\mathbf{Q}_k + \mathbf{f}_m\mathbf{e}_m^T\mathbf{Q}_k,\\ &= \mathbf{X}_m\mathbf{Q}_k\mathbf{H}_k^+ + \beta_{k+1}^+\mathbf{X}_m\mathbf{Q}_m\mathbf{e}_{k+1}\mathbf{e}_k^T + \mathbf{f}_m\mathbf{e}_m^T\mathbf{Q}_k,\\ &= \mathbf{X}_m\mathbf{Q}_k\mathbf{H}_k^+ + \mathbf{f}_k^+\mathbf{e}_k^T,\end{aligned} \tag{6.2}$$

where $\mathbf{f}_k^+ \equiv \beta_{k+1}^+\mathbf{X}_m\mathbf{Q}_m\mathbf{e}_{k+1} + (\mathbf{e}_m^T\mathbf{Q}_k\mathbf{e}_k)\mathbf{f}_m$. Thus, an IRA-iteration may be viewed as a truncated QR algorithm. One cycle of the iteration is illustrated in Figures 6.1– 6.3.

**7. Explicit and Implicit Restarting.** We present a striking example that compares the ERA and IRA iterations. Let $\mathbf{A} \in \mathbf{R}^{10\times 10}$ be zero everywhere except for diagonal elements

$$\alpha_{11} = 1, \alpha_{22} = 1, \alpha_{33} = 0, \alpha_{44} = 0, \alpha_{ii} = (5 - i)\cdot 10^{-1}, \text{ for } i = 1, \ldots, 6,$$

and ones on the subdiagonal. Suppose that the vector $\mathbf{e}_1$ is used to start both explicit and implicitly restarted Arnoldi algorithms with $k = 2$ and $m = 4$ with the interest to compute the two eigenvalues equal to one. Using an exact shift strategy, an IRA iteration computes the approximate partial real Schur decomposition $\mathbf{A}\mathbf{Q}_2 \approx \mathbf{Q}_2\mathbf{R}_2$, where

$$\mathbf{R}_2 \approx \left[\begin{array}{cc} .94919 & .95789 \\ -2.6952\cdot 10^{-3} & 1.0508\end{array}\right],$$

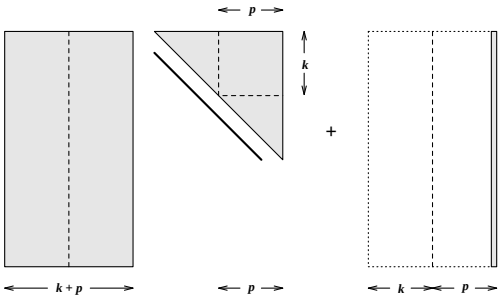


FIG. 6.1. *The set of rectangles represents the matrix equation* $\mathbf{X}_m \mathbf{H}_m + \mathbf{f}_m \mathbf{e}_m^T$ *of an Arnoldi reduction. The unshaded region on the right is a zero matrix of* $m-1$ *columns.*

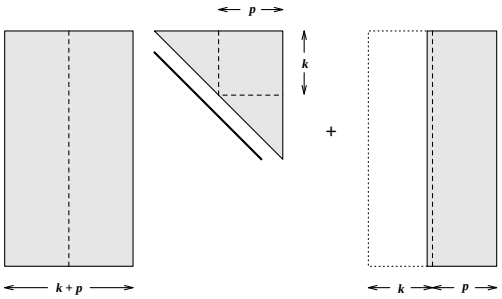


FIG. 6.2. *After performing* $m-r$ *implicitly shifted* QR *steps on* $\mathbf{H}_m$*, the middle set of pictures illustrates* $\mathbf{X}_m \mathbf{Q}_m \mathbf{H}_m^+ + \mathbf{f}_m \mathbf{e}_m^T \mathbf{Q}_m$*. The last* $p$ *columns of* $\mathbf{f}_m \mathbf{e}_m^T \mathbf{Q}_m$ *are nonzero because of the* QR *iteration.*

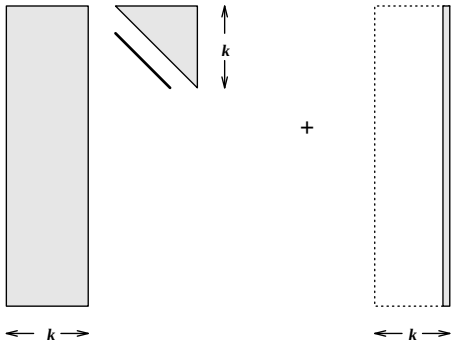


FIG. 6.3. *After discarding the last* $m-r$ *columns, the final set represents* $\mathbf{X}_m \mathbf{Q}_k \mathbf{H}_k^+ + \mathbf{f}_k^+ \mathbf{e}_k^T$ *of a length* $r$ *Arnoldi factorization.*

with eigenvalues equal to $1 \pm i1.129168612228906 \cdot 10^{-8}$. The number of iterations needed was four, and a total of ten matrix vector products was computed.

However, the algorithm in Table 5.1 stagnates if the expansion coefficients are chosen as originally proposed by Saad, as explained in §5. In fact, the starting vector $\mathbf{e}_1$ is computed during every cycle of the ERA iteration! During every iteration,

$$\mathbf{H}_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

is computed. The MATLAB function `EIG` computes the two eigenvectors

$$\begin{aligned}\mathbf{s}_1^T &= \begin{bmatrix} 0 & .57735 & .57735 & .57735 \end{bmatrix},\\ \mathbf{s}_2^T &= -\mathbf{s}_1^T + 1.8 \cdot 10^{-18}\mathbf{e}_1^T\end{aligned}$$

corresponding to the two eigenvalues equal to one. Choosing $\mathbf{y}_4$ to be an unit vector in the linear span of $\begin{bmatrix} \|\mathbf{A}\mathbf{X}_4\mathbf{s}_1 - \mathbf{X}_4\mathbf{s}_1\| & \|\mathbf{A}\mathbf{X}_4\mathbf{s}_2 - \mathbf{X}_4\mathbf{s}_2\| & 0 & 0 \end{bmatrix}^T$ gives that $\mathbf{X}_4\mathbf{S}_4\mathbf{y}_4 = \pm\mathbf{e}_1$.

The explanation is simple enough. Although $\mathbf{e}_1$ is orthogonal to the eigenspace associated with the eigenvalue one of $\mathbf{A}$, it is not orthogonal to the invariant subspace associated with the unit eigenvalue. This explains why the IRA-iteration converges while the ERA iteration does not.

The major drawback of using a linear combination of the eigenvectors of $\mathbf{H}_m$ is that they may form a poor choice for the starting vector. If $\mathbf{H}_m$ is defective, there might not be enough eigenvectors associated with the wanted eigenvalues. A pair of approximate eigenvectors is produced that are aligned to working precision. On the other hand, using an expansion in terms of the Schur vectors of $\mathbf{H}_m$ gives a "richer" starting vector. Theorem 8.1 gives that the IRA iteration implicitly uses a Schur basis of $\mathbf{H}_m$.

Golub and Wilkinson [12] examine the many practical difficulties involved when computing invariant subspaces. They conclude that working with Schur vectors is a better behaved numerical process. Within the context of subspace iteration, Stewart [41] also arrives at the same conclusion.

**8. Characterzing an IRA Iteration.** We present three theorems that explain the behavoir of an IRA iteration in exact arithmetic. The first theorem is a generalization of Lemma 3.10 proved by Sorensen [39] and indicates what occurs when a certain choice of shifts is used during an IRA iteration. There are two major differences. The first is that there is no assumption on the existence of a basis of eigenvectors for the desired invariant subspace. Only a Schur basis is used. The second is that we make no assumptions on the multiplicities of the shifts applied.

THEOREM 8.1. *Suppose $\mathbf{H}_m \in \mathbf{R}^{m\times m}$ is an unreduced upper Hessenberg matrix corresponding to a length $m$ Arnoldi reduction $\mathbf{A}\mathbf{X}_m = \mathbf{X}_m\mathbf{H}_m + \mathbf{f}_m\mathbf{e}_m^T$ and that the eigenvalues of $\mathbf{H}_m$ are in the partition*

$$\{\theta_1, \ldots, \theta_k\} \cup \{\theta_{k+1}, \ldots, \theta_m\}.$$

*Assume that the complex conjugate pairs of eigenvalues are kept together; $\theta_i = \bar{\theta}_j$ implies that $i, j \leq k$ or $i, j > k$ and that $\mathbf{H}_m\mathbf{Z}_k = \mathbf{Z}_k\mathbf{T}_k$ is a partial real Schur decomposition, where the eigenvalues of $\mathbf{T}_k$ are $\theta_1, \ldots, \theta_k$.*

*If $m - k$ QR steps are performed with the eigenvalues $\theta_{k+1}, \ldots, \theta_m$ producing an orthogonal matrix $\mathbf{Q}_m \in \mathbf{R}^{m\times m}$ such that*

$$\mathbf{H}_m\mathbf{Q}_m = \mathbf{Q}_m \begin{bmatrix} \mathbf{H}_{11}^+ & \mathbf{H}_{12}^+ \\ 0 & \mathbf{H}_{22}^+ \end{bmatrix}, \tag{8.1}$$

*the eigenvalues of $\mathbf{H}_{11}^+$ are $\theta_1, \ldots, \theta_k$ and the first $k$ columns of $\mathbf{Q}_m$ span the associated invariant subspace.*

*Moreover, the updated starting vector $\mathbf{X}_m\mathbf{Q}_m e_1 \in \mathcal{R}(\mathbf{X}_m\mathbf{Z}_k)$, and*

$$\mathbf{A}\mathbf{X}_m\mathbf{Q}_k = \mathbf{X}_m\mathbf{Q}_k\mathbf{H}_{11}^+ + (\mathbf{e}_m^T\mathbf{Q}_k\mathbf{e}_k)\mathbf{f}_m\mathbf{e}_k^T, \tag{8.2}$$

*is the updated Arnoldi reduction of length $k$.*

*Proof.* A result by Miminis and Paige [23, pages 391–395] proves equation (8.1). They prove that if $m-k$ QR steps are performed, the matrix equation (8.1) results if and only if the $m-k$ shifts are eigenvalues of $\mathbf{H}_m$, regardless of their multiplicity.

Let $\mathbf{Q}_k = \mathbf{Q}_m \left[ \begin{array}{cccc} \mathbf{e}_1 & \mathbf{e}_2 & \cdots & \mathbf{e}_k \end{array} \right]$, where $\mathbf{H}_m\mathbf{Q}_k = \mathbf{Q}_k\mathbf{H}_{11}^+$. Let $\mathbf{H}_m\mathbf{Z}_k = \mathbf{Z}_k\mathbf{T}_k$, be a real Schur decomposition where the eigenvalues of $\mathbf{H}_{11}^+$ and $\mathbf{T}_k$ are the same. Thus, the columns of $\mathbf{Q}_k$ and $\mathbf{Z}_k$ span the same invariant subspace. It follows that

$$\mathbf{X}_m\mathbf{Q}_m\mathbf{e}_1 = \mathbf{X}_m\mathbf{Q}_k\mathbf{e}_1 = \mathbf{X}_m\mathbf{Z}_k\mathbf{Z}_k^T\mathbf{Q}_k\mathbf{e}_1 \equiv \mathbf{X}_m\mathbf{Z}_k\mathbf{y}_m,$$

where $\mathbf{y}_m = \mathbf{Z}_k^T\mathbf{Q}_k\mathbf{e}_1$. Postmultiply the length $m$ Arnoldi reduction with $\mathbf{Q}_k$ and use equation (6.2) to obtain equation (8.2), since $\beta_{k+1}^+ = 0$. ∎

The restriction that keeps the complex conjugate pairs of eigenvalues together is needed only so that the iteration may be done in real arithmetic.

Using the exact shifting strategy during the IRA iteration replaces the starting vector with a linear combination of the wanted approximate Schur vectors. The ERA iteration also has the same goal, but the IRA iteration performs this replacement implicitly in a stable fashion by using a Schur basis of $\mathbf{H}_m$. Moreover, the IRA iteration avoids the need to restart the next reduction from scratch.

The second result shows that the polynomial implicitly applied by an IRA iteration using exact shifts is of minimal degree when we wish to restart an Arnoldi reduction with a vector that is a linear combination of wanted spectral information in $\mathbf{H}_m$.

THEOREM 8.2. *Assume the same hypothesis of Theorem 8.1 with the addition that the eigenvalues of $\mathbf{H}_m$ are distinct. Let*

$$\psi(\lambda) = \prod_{j=k+1}^{m} (\lambda - \theta_j),$$

*and denote the Ritz vectors by $\mathbf{z}_j = \mathbf{X}_m\mathbf{s}_j$, where $\mathbf{H}_m\mathbf{s}_j = \mathbf{s}_j\theta_j$.*

*If $\hat{\mathbf{x}}_1 \in Span\{\mathbf{z}_1, \ldots, \mathbf{z}_k\}$, then for some polynomial $\phi(\lambda)$ of degree not exceeding $m-1$,*

$$\hat{\mathbf{x}}_1 = \phi(\mathbf{A})\mathbf{x}_1,$$

*where $\phi(\lambda) = \psi(\lambda)\chi(\lambda)$ for some polynomial $\chi(\lambda)$ of degree at most $k-1$.*

*Proof.* Let $\mathbf{z}_j \in \mathcal{K}_m(\mathbf{A}, \mathbf{x}_1)$. Then, for every $j$, there is polynomial $p_j(\lambda)$ of degree not exceeding $m-1$ such that $\mathbf{z}_j = p_j(\mathbf{A})\mathbf{x}_1$. Thus $\hat{\mathbf{x}}_1 = \phi(\mathbf{A})\mathbf{x}_1$ for some polynomial $\phi(\lambda)$ of degree not exceeding $m-1$. An easy consequence of equation (6.1) is that

$$\hat{\mathbf{x}}_1 = \phi(\mathbf{A})\mathbf{x}_1 = \phi(\mathbf{A})\mathbf{X}_m\mathbf{e}_1 = \mathbf{X}_m\phi(\mathbf{H}_m)\mathbf{e}_1.$$

Expand $\mathbf{e}_1 = \mathbf{s}_1\xi_1 + \cdots + \mathbf{s}_m\xi_m$ and hence $\phi(\mathbf{H}_m)\mathbf{e}_1 = \mathbf{s}_1\phi(\theta_1)\xi_1 + \cdots + \mathbf{s}_m\phi(\theta_m)\xi_m$. Since $\hat{\mathbf{x}}_1 \in \mathrm{Span}\{\mathbf{z}_1, \ldots, \mathbf{z}_k\}$, it follows that $\phi(\theta_j)\xi_j = 0$ for $j = k+1, \ldots, n$. Denote the left eigenvectors of $\mathbf{H}_m$ by $\mathbf{v}_j$ indexed so that $\mathbf{v}_j^H\mathbf{H}_m = \mathbf{v}_j^H\theta_j$. Since the eigenvalues of $\mathbf{H}_m$ are distinct, the biorthogonality of the left and right eigenvectors of $\mathbf{H}_m$ gives that $\mathbf{v}_j^H\mathbf{e}_1 = \mathbf{v}_j^H\mathbf{s}_j\xi_j$ and $\mathbf{v}_j^H\mathbf{s}_j \neq 0$ for $j = 1, \ldots, m$. Standard results on unreduced upper Hessenberg matrices give that $\mathbf{v}_j^H\mathbf{e}_1 \neq 0$. Hence $\xi_j \neq 0$, and so $\phi(\theta_j) = 0$ for $j = k+1, \ldots, m$. Thus $\psi(\lambda)$ must be a divisor of $\phi(\lambda)$, and the theorem is proved. ∎

If the goal of restarting is to select an improved starting vector consisting of only wanted Ritz vectors, Theorem 8.2 states that unwanted components might be

introduced if the degree of the polynomial is greater than $m-k$. Since any starting vector that is a linear combination of the columns of $\mathbf{X}_m$ is equal to $\phi(\mathbf{A})\mathbf{x}_1$ for some polynomial $\psi(\cdot)$ of degree not exceeding $m-1$, any linear combination of wanted Ritz vectors as in §5 is possibly introducing unwanted components.

The following result shows that the Krylov space associated with the subsequent Arnoldi reduction computed after an implicit restart with exact shifts also has some interesting subspaces. It is a slight generalization of Theorem 3 proved by Morgan [25].

THEOREM 8.3. *Assume the same hypothesis and notation as Theorem 8.1 with the additional hypothesis that* $\mathbf{f}_m \neq \mathbf{0}$. *Suppose that* $m-k=p$ QR *steps are performed with the eigenvalues* $\theta_{k+1},\ldots,\theta_m$. *If* $\mathbf{A}\mathbf{X}_m^+ = \mathbf{X}_m^+\mathbf{H}_m^+ + \mathbf{f}_m^+\mathbf{e}_m^T$ *is the length* $m$ *Arnoldi reduction that results from extending the compressed reduction of equation (8.2) and* $\mathbf{H}_m^+$ *is unreduced, then*

$$\mathcal{R}(\mathbf{X}_m^+) = \mathcal{R}(\mathbf{X}_m\mathbf{Q}_k) \cup \mathit{Span}\{\mathbf{A}\mathbf{z}_j,\ldots,\mathbf{A}^p\mathbf{z}_j\} \tag{8.3}$$

*holds for each Ritz vector* $\mathbf{z}_j = \mathbf{X}_m\mathbf{s}_j$, *where* $\mathbf{H}_m\mathbf{s}_j = \mathbf{s}_j\theta_j$ *for* $j=1,\ldots,k$.

*Proof.* Partition the eigenvalues of $\mathbf{H}_m$ as in the hypothesis of Theorem 8.1. Let $(\mathbf{s}_j,\theta_j)$ be an eigenpair for $\mathbf{H}_m$, where $\|\mathbf{s}_j\|=1$, and set $\mathbf{z}_j = \mathbf{X}_m\mathbf{s}_j$. Define $\mathbf{x}_{m+1} \equiv \mathbf{f}_m/\|\mathbf{f}_m\|$ and $\mathbf{X}_m^+\mathbf{e}_j \equiv \mathbf{x}_j^+$ for $j=1,\ldots,m$. Standard results give that $\mathbf{x}_{k+1}^+ = \psi_k(\mathbf{A})\mathbf{x}_1^+$ for some polynomial $\psi_k(\lambda)$ of degree $k$.

Note that by equation (8.2) of Theorem 8.1, we have $\mathbf{x}_{k+1}^+ = \mathbf{x}_{m+1}$. It also follows that $\mathbf{A}^i\mathbf{x}_{k+1}^+ = \mathbf{A}^i\psi_k(\mathbf{A})\mathbf{x}_1^+ \in \mathcal{K}_{k+i+1}(\mathbf{A},\mathbf{x}_1^+)$ for $i=1,\ldots,m-k-1$, which implies that

$$\mathrm{Span}\{\mathbf{X}_m\mathbf{Q}_k, \mathbf{x}_{k+1}^+,\ldots,\mathbf{A}^{m-k-1}\mathbf{x}_{k+1}^+\} \subset \mathcal{R}\{\mathbf{X}_m^+\}. \tag{8.4}$$

We now show that these two sets share the same dimension.

Suppose that $\mathbf{X}_m\mathbf{Q}_k\mathbf{y}_1 + \mathbf{K}_{m-k}(\mathbf{A},\mathbf{x}_{k+1}^+)\mathbf{y}_2 = 0$ for some $\mathbf{y} \equiv \begin{bmatrix} \mathbf{y}_1^T & \mathbf{y}_2^T \end{bmatrix}^T \in \mathbf{R}^m$. Thus, there exists a polynomial $\psi(\lambda)$ of degree less than $m$ so that $\psi(\mathbf{A})\mathbf{x}_1^+ = 0$. However, since $\mathbf{H}_m^+$ is unreduced, the grade of $\mathbf{x}_1^+$ is at least $m$, and hence $\mathbf{y} \equiv \mathbf{0}$, which implies that the two sets in equation (8.4) are equal.

Using mathematical induction, we show that

$$\mathbf{A}^i\mathbf{z}_j \in \mathrm{Span}\{\mathbf{z}_j, \mathbf{x}_{k+1}^+,\ldots,\mathbf{A}^{i-1}\mathbf{x}_{k+1}^+\},$$

for $i=1,\ldots,m-k$. From the length $m$ Arnoldi reduction, it follows that

$$\mathbf{A}\mathbf{z}_j = \mathbf{z}_j\theta_j + \mathbf{f}_m(\mathbf{e}_m^T\mathbf{s}_j) = \mathbf{z}_j\theta_j + \mathbf{x}_{m+1}(\mathbf{e}_m^T\mathbf{s}_j)\|\mathbf{f}_m\| \in \mathrm{Span}\{\mathbf{z}_j, \mathbf{x}_{k+1}^+\},$$

establishing the base case. Suppose that the result is true for positive integers $i-1$. The inductive hypothesis implies that

$$\begin{aligned}\mathbf{A}^i\mathbf{z}_j &\in \mathbf{A}\mathbf{A}^{i-1}\mathbf{z}_j \\ &\in \mathbf{A}\mathrm{Span}\{\mathbf{z}_j, \mathbf{x}_{k+1}^+,\ldots,\mathbf{A}^{i-2}\mathbf{x}_{k+1}^+\} \\ &\in \mathrm{Span}\{\mathbf{z}_j, \mathbf{x}_{k+1}^+,\ldots,\mathbf{A}^{i-1}\mathbf{x}_{k+1}^+\},\end{aligned}$$

and the desired result follows. Now, since $\mathbf{z}_j \in \mathcal{R}\{\mathbf{X}_m\mathbf{Q}_k\}$ and $\mathbf{x}_{k+1}^+ = \psi_k(\mathbf{A})\mathbf{x}_1^+$, it follows from the established equality of the two sets in equation (8.4) that

$$\mathrm{Span}\{\mathbf{X}_m\mathbf{Q}_k, \mathbf{A}\mathbf{z}_j,\ldots,\mathbf{A}^{m-k}\mathbf{z}_j\} \subset \mathcal{R}\{\mathbf{X}_m^+\}. \tag{8.5}$$

If we use a similar argument to the one that followed equation (8.4), the two sets in equation (8.5) are equal. The first conclusion of the theorem in equation (8.3) is proved. □

In particular, if the eigenvectors $\mathbf{s}_1, \ldots, \mathbf{s}_k$ of $\mathbf{H}_m$ are linearly independent, then

$$\mathcal{R}(\mathbf{X}_m^+) = \mathcal{K}_p(\mathbf{A}, \mathbf{z}_j) \cup_{i \neq j} \text{Span}\{\mathbf{z}_i\} \tag{8.6}$$

for $j = 1, \ldots, k$. Morgan concludes that the Krylov subspace of length $m$ generated during each cycle of an IRA iteration with exact shifts contains all the Krylov subspaces of dimension $p+1$ generated from a wanted Ritz vector, since

$$\mathcal{K}_{p+1}(\mathbf{A}, \mathbf{z}_j) \subset \mathcal{K}_m(\mathbf{A}, \mathbf{X}_m^+ \mathbf{e}_1) \equiv \mathcal{R}(\mathbf{X}_m^+).$$

**9. Polynomial Accelerations Techniques.** Suppose $\mathbf{A}$ is diagonalizable with eigenpairs $(\mathbf{u}_j, \lambda_j)$ for $j = 1, \ldots, n$. If $\psi(\cdot)$ is some polynomial and we expand the current starting vector $\mathbf{x}_1$ in terms of the basis of eigenvectors, then

$$\psi(\mathbf{A})\mathbf{x}_1 = \mathbf{u}_1 \psi(\lambda_1)\zeta_1 + \cdots + \mathbf{u}_n \psi(\lambda_n)\zeta_n. \tag{9.1}$$

Assuming that the eigenpairs $(\mathbf{u}_i, \lambda_i)$ are ordered so that the wanted $k$ ones are at the beginning of the expansion, we seek a polynomial such that

$$\max_{i=k+1,\ldots,n} |\psi(\lambda_i)| < \min_{i=1,\ldots,k} |\psi(\lambda_i)|. \tag{9.2}$$

A good polynomial $\psi(\lambda)$ acts as a *filter*. Components in the direction of unwanted eigenvectors are damped, or, equivalently, components in the direction of wanted eigenvectors are amplified.

The acceleration techniques and hybrid methods presented by Saad in Chapter 7 of [36] attempt to improve the ERA iteration introduced in §5 by approximately solving the min-max problem of equation (9.2). Motivated by Manteuffel's scheme [20], Saad first proposed the use of Chebyshev polynomials in [34]. A Chebyshev polynomial $\psi(\mathbf{A})$ on an ellipse containing the unwanted Ritz values is applied to the restart vector in an attempt to accelerate convergence of the orginal ERA iteration. The polynomial is applied with the use of the familiar three-term recurrence.

Table 9.1 outlines the procedure. It is called a deflated iteration because a partial Schur decomposition is incrementaly built. We refer to the orthogonlization process in line 2 as *locking*. At each iteration, the Ritz pairs of the unlocked portion of the Arnoldi reduction are ordered with respect some criterion. For example, if the eigenvalues of largest magnitude are desired, the Ritz pair of interest is the one associated with the Ritz that satisfies this criterion. In order to compute in real arithmetic, the procedure outlined in §2 is employed at line 2 if $\theta_1$ is not a real number.

**10. Numerical Results.** Lehoucq and Scott [17] presented a software survey of large-scale eigenvalue methods and comparative results. The Arnoldi-based software included the following three packages, which are available either in the public domain or under licence. These are the `ARNCHEB` package [4], the `ARPACK` [19] software package, and the Harwell Subroutine Library code `EB13` [37].

The `ARNCHEB` package provides the subroutine `ARNOL`, which implements an explicitly restarted Arnoldi iteration. The code is based on the algorithm given in Table 9.1 without the use of locking. It uses Chebyshev polynomial acceleration.

The Harwell Subroutine Library code `EB13` implements the algorithm given in Table 9.1 and also uses Chebyshev polynomial acceleration.

TABLE 9.1
*A deflated polynomial accelerated* ERA *iteration*

- Build a length $m$ Arnoldi reduction. Set $j = 0$ and define $\mathbf{Q}_0$ and $\mathbf{T}_0$ as empty matrices.
- *Iteration:*
  1. Compute the Ritz pair of interest $(\mathbf{z}_1, \theta_1)$.
  2. Check to see if $(\mathbf{z}_1, \theta_1)$ is an acceptable approximation. If so, increment $j$ and compute an unit vector $\mathbf{q}_j$ in the direction of $\mathbf{z}_1 - \mathbf{Q}_{j-1}\mathbf{t}_j$ where $\mathbf{t}_j = \mathbf{Q}_{j-1}^T \mathbf{z}_1$. Set $\mathbf{Q}_j = \begin{bmatrix} \mathbf{Q}_{j-1} & \mathbf{q}_j \end{bmatrix}$ and $\mathbf{T}_j = \begin{bmatrix} \mathbf{T}_{j-1} & \mathbf{t}_j \\ 0 & \theta_1 \end{bmatrix}$.
  3. If $j = k$, exit the *Iteration*.
  4. Compute the Arnoldi reduction $\mathbf{A}\begin{bmatrix} \mathbf{Q}_j & \mathbf{X}_{m-j} \end{bmatrix} = \begin{bmatrix} \mathbf{Q}_j & \mathbf{X}_{m-j} \end{bmatrix} \begin{bmatrix} \mathbf{T}_j & \mathbf{M}_j \\ & \mathbf{H}_{m-j} \end{bmatrix} + \mathbf{f}_m \mathbf{e}_m^T$ of length $m$ with starting vector $\psi(\mathbf{A})\mathbf{X}_{m-j}\mathbf{e}_1$ orthogonal to the $\mathcal{R}(\mathbf{Q}_j)$.

The `ARPACK` software package provides subroutine `DNAUPD` that implements an implicitly restarted Arnoldi iteration.

We present results for the two-dimensional model convection-diffusion problem

$$-\Delta \mathbf{u}(x, y) + \rho \nabla \cdot \nabla \mathbf{u}(x, y) = \lambda \mathbf{u}(x, y),$$

on the unit square $[0, 1] \times [0, 1]$ with zero boundary data. Here, $\rho$ represents the convection and is a real number. The problem is discretized by using centered finite differences. The eigenvalues and eigenvectors of the resulting matrix are known explicitly. This feature allows us to check the accuracy of our results.

The resulting matrix is interesting because of the following properties:

- Many of the eigenvalues have multiplicity two. It may be shown that if $|\rho| \leq \sqrt{n}$, the eigenvalues are all real and the matrix is diagonalizable.
- As the mesh size decreases, the relative separation of all the eigenvalues decreases. All the eigenvalues are contained within the interval (0, 8).
- As $\rho$ increases, so does the nonnormality of the matrix.

We computed $k = 6$ eigenpairs of largest real part for a range of values of $\rho$ and for orders up to $n = 10,000$. Tables 10.1 and 10.2 display the results of experiments run on an IBM RS/6000 3BT in double-precision arithmetic for two specific matrices. The column heading `WHICH` gives the portion of the spectrum the respective code was asked to compute. The eigenvalues of largest real part are also those of largest modulus. They are respectively denoted by `WHICH = 'LR' and 'LM'`. The length of the Arnoldi reduction generated is denoted by $m$. All the Ritz pairs computed were checked and gave residuals of order $\mathcal{O}(10^{-12})$ with at least seven digits of accuracy for the Ritz values. The numerical orthogonality of the Arnoldi vectors was also checked.

The results show that the implementation of the IRA iteration significantly reduces the total number of matrix vector products required. For many large-scale eigenvalue problems, the dominant cost is that of performing the matrix-vector products. For these two examples, `ARPACK` reliably computed the multiplicities. Changing `WHICH` has the effect of modifying the restart parameters. For further information and other experiments, we refer the reader to [17].

**11. Convergence of an IRA Iteration.** Sorensen gives a convergence theorem [39, pages 369–370] for an IRA iteration where the polynomial applied at every restart is fixed. The main result of this section gives conditions that determine the

TABLE 10.1
*CPU times (in seconds) and matrix-vector products for the CDDE matrix with $\rho = 10$ of order 2500. († denotes that one or more of the requested eigenvalues was missed; ∗ denotes that code did not converge within 4000 m-dimensional matrix-vector products).*

| | | Subspace Dimension $m$ | |
|---|---|---|---|
| Algorithm | WHICH | 18 | 36 |
| ARNCHEB | LR | † | † |
| ARPACK | LM | 8.6/620 | 12/694 |
| ARPACK | LR | 8.3/602 | 11/613 |
| EB13 | LM | † | 46/3383 |
| EB13 | LR | 41/12178 | ∗ |

TABLE 10.2
*CPU times (in seconds) and matrix-vector products for the CDDE matrix with $\rho = 15$ of order 10000 (∗ denotes that code did not converge within 4000 m-dimensional matrix-vector products)*

| | | Subspace Dimension $m$ | |
|---|---|---|---|
| Algorithm | WHICH | 18 | 36 |
| ARNCHEB | LR | ∗ | ∗ |
| ARPACK | LM | 71/1123 | 103/1398 |
| ARPACK | LR | 61/991 | 80/1095 |
| EB13 | LM | 727/20004 | 436/7263 |
| EB13 | LR | 1251/74107 | ∗ |

convergence of an IRA iteration where the polynomial applied at every restart may vary. Our approach exploits the fact that an IRA iteration is a curtailed QR iteration and thus allows us to use this well-established convergence theory on upper Hessenberg matrices. As mentioned in §2.1, computing a partial real Schur decomposition corresponding to a small subset of the eigenvalues of $\mathbf{A}$ is the objective.

Let $l = p_1 + \cdots + p_r$, where $r$ designates how many restarts are performed, $p_i$ the degree of the polynomial applied at the $i$th restart, and $\mathbf{Z}_m^{(l)}$ the accumlation of all the orthogonal matrices applied through the QR iterations. Using the relationships derived through equation (6.2), we write the length $k$ Arnoldi reduction as

$$\mathbf{A}\mathbf{X}_m\mathbf{Z}_k^{(l)} = \mathbf{X}_m\mathbf{Z}_k^{(l)}\mathbf{H}_k^{(l)} + \mathbf{f}_k^{(l)}\mathbf{e}_k^T, \tag{11.1}$$

where $\mathbf{Z}_k^{(l)} = \mathbf{Z}_m^{(l)} \begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \cdots & \mathbf{e}_k \end{bmatrix}$ and $\mathbf{X}_m$ denotes the initial reduction to Hessenberg form. Thus, we may examine the convergence of an IRA iteration by studying the convergence of $\mathbf{X}_m\mathbf{Z}_k^{(l)}$ to an invariant subspace of dimension $k$. Since the Implicit Q theorem (see §4) gives that $\mathbf{H}_k^{(l)}$ is the leading principal matrix of order $k$ of $\mathbf{H}_n^{(l)}$ that would be obtained if we performed the *full* QR algorithm on the *full* Hessenberg matrix $\mathbf{H}_n$, we study the convergence of the shifted QR iteration.

A convergence theory for the shifted QR iteration was presented by Watkins and Elsner [43] within the more general framework of generic *GR algorithms*. A GR algorithm is an iterative procedure such as the QR iteration where the QR factorization is replaced with a decomposition of the form $\mathbf{G}_n\mathbf{R}_n = \mathbf{H}_n - \tau\mathbf{I}_n$, where $\mathbf{R}_n$ is upper triangular and $\mathbf{G}_n$ is a nonsingular matrix.

THEOREM 11.1. *Let $\mathbf{H}_n \in \mathbf{R}^{n\times n}$ be an unreduced upper Hessenberg matrix and*

*$\Psi(\lambda)$ be a polynomial. Order the eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ of $H$ so that $|\Psi(\lambda_1)| \geq |\Psi(\lambda_2)| \geq \cdots \geq |\Psi(\lambda_n)|$.*

*Let $\mathbf{H}_n\mathbf{Z}_n = \mathbf{Z}_n\mathbf{T}_n$ be a real Schur decomposition where the first $k$ columns of $\mathbf{Z}_n$ span an invariant subspace corresponding to the eigenvalues $\lambda_1, \ldots, \lambda_k$. Suppose $k$ is a positive integer less than $n$ such that $\rho_k \equiv |\Psi(\lambda_{k+1})|/|\Psi(\lambda_k)| < 1$.*

*If a sequence of shifts $\{\tau_i\}_{i=1}^{l}$ has the properties that $\prod_{i=1}^{l}\tau_i$ is a real number and*

$$\begin{aligned} \mathcal{P}_l(\lambda_i) \equiv (\lambda_i - \tau_1)\cdots(\lambda_i - \tau_l) &\rightarrow \Psi(\lambda_i), \quad i = k+1, \ldots, n \\ \mathcal{P}_l(\lambda_i) &\neq 0, \quad i = 1, \ldots, k, \end{aligned}$$

*as $l \rightarrow \infty$, the* QR *iteration computes $\mathbf{H}_n\mathbf{Z}_n^{(l)} = \mathbf{Z}_n^{(l)}\mathbf{H}_n^{(l)}$, where*

$$\mathbf{H}_n^{(l)} \equiv \begin{bmatrix} \mathbf{H}_k^{(l)} & \mathbf{M}_k^{(l)} \\ \beta_{k+1}^{(l)}\mathbf{e}_1\mathbf{e}_k^T & \bar{\mathbf{H}}_{n-k}^{(l)} \end{bmatrix},$$

*and $\mathbf{Z}_n^{(l)}$ is an orthogonal matrix such that for every value of $\hat{\rho}_k$ satisfying $\rho_k < \hat{\rho}_k < 1$, there exists a constant $C$ such that $|\beta_{k+1}^{(l)}| \leq C(\hat{\rho}_k)^l$ and*

$$dist(\mathbf{Z}_k, \mathbf{Z}_k^{(l)}) \leq C(\hat{\rho}_k)^l,$$

*where $\mathbf{Z}_k^{(l)}$ contains the first $k$ columns of $\mathbf{Z}_n^{(l)}$.*

*Proof.* See Theorems 5.4 and 6.2 of Watkins and Elsner [43]. □

If we partition the eigenvalues of $\mathbf{A}$ so that $\lambda_1, \ldots, \lambda_k$ are the sought-after eigenvalues, then

$$\Psi(\lambda) = \prod_{i=k+1}^{n}(\lambda - \lambda_i), \tag{11.2}$$

is an example of the polynomial used by the theorem. The theorem gives the convergence rate of $\beta_{k+1}^{(l)}$ to zero, given a shifting strategy. Note that by the Implicit Q theorem (see §4) and equation (11.1), $\|\mathbf{f}_k^{(l)}\| = \beta_{k+1}^{(l)}$. The shifting strategy has the effect of replacing the starting vector, thereby restarting the reduction. Thus, the convergence of an IRA iteration is established.

The distance between the subspaces [6, 10] $\mathcal{R}(\mathbf{Z}_k)$ and $\mathcal{R}(\mathbf{Z}_k^{(l)})$ may be shown to be equal to $\sqrt{1 - \|\mathbf{Z}_k^T\mathbf{Z}_k^{(l)}\|^2}$. For increasing values of $l$, the approximate Schur basis vectors contained in $\mathbf{Z}_k^{(l)}$ span $\mathcal{R}(\mathbf{Z}_k)$. Thus, the $\text{dist}(\mathbf{Z}_k, \mathbf{Z}_k^{(l)}) \rightarrow 0$, and the eigenvalues of $\mathbf{H}_k^{(l)}$ tend to $\lambda_1, \ldots, \lambda_k$. It follows from the theorem that for all values of $k$ such that $\rho_k < 1$, the $k$th subdiagonal element of $\mathbf{H}_n^{(l)}$ tends to zero. The hypothesis on the product of the shifts ensures that if one is applied with a nonzero imaginary part, its complex conjugate is also a shift.

The theorems proved by Watkins and Elsner in [43] identify the convergence of the QR algorithm with that of simultaneous iteration, or subspace iteration. Parlett [29] presents the first set of comprehensive sufficient conditions for convergence of the QR iteration on Hessenberg matrices, while a portion of the paper by Parlett and Poole [31] considers a geometric convergence theory for Hessenberg matrices.

**12. Practical Considerations.** We have shown a direct connection between the IRA and QR iterations. With this connection, we believe that reliable general-purpose software for the large-scale eigenvalue problem is possible. The practical QR algorithm [21] resulted when deflation rules and practical shifting strategies were incorporated. These techniques are extremely important for the convergence and stability of the iteration. They have contributed to the emergence of the practical QR algorithm as the method of choice for computing the eigensystem of dense matrices. In particular, the deflation rules allow the QR iteration to compute multiple and clustered eigenvalues. The reader is referred to [18] for a detailed study on deflation strategies for an IRA iteration. The remainder of this section discusses shifting strategies. We remark that the shifting strategy of the practical QR algorithm cannot by employed because it requires the full reduction to upper Hessenberg form at every iteration.

Although useful for characterzing the convergence of an IRA iteration, Theorem 11.1 gives no indication on how well $\mathcal{P}_l(\cdot)$ must approximate the polynomial $\Psi(\lambda)$, let alone a practical shifting strategy. This is the subject of current research. We have examined one particular choice of polynomial, namely, one constructed from the unwanted Ritz values at every iteration. Other interesting strategies include the roots of Chebyshev polynomials [34], Harmonic Ritz values [24, 26, 28, 38], the roots of Leja polynomials [3, 5], and the roots of least squares polynomials [35]. In particular, the Harmonic Ritz values have been used to estimate the interior eigenvalues of $\mathbf{A}$.

There also remains the interesting question of how many shifts $p_i$ to apply per iteration. For example, if $\Psi(\lambda) = \lambda^n$ with $|\lambda_k| > |\lambda_{k+1}|$, then using zero shifts gives that $\beta_{k+1}^{(l)}$ goes to zero with the convergence rate of $|\lambda_{k+1}/\lambda_k|$. It is well known that performing subspace iteration on a subspace of dimension larger than the number of eigenvalues required typically leads to improved convergence rates; see the paper of Duff and Scott [9] for a discussion and further references.

For more general shifting strategies, we can expect $\beta_{k+1}^{(l)}$ to converge with the convergence rate of

$$\frac{\max\limits_{k+1\le i\le n} |\mathcal{P}_l(\lambda_i)|}{\min\limits_{1\le i\le k} |\mathcal{P}_l(\lambda_i)|} \approx \frac{|\Psi(\lambda_{k+1})|}{|\Psi(\lambda_k)|}. \tag{12.1}$$

For example, if we use the unwanted Ritz values as shifts and $\Psi(\lambda)$ is defined as in equation (11.2), the convergence rate is approximately given by the ratio in the left hand side of equation (12.1). Since the ratio is a complicated function involving the shifts applied, it is not obvious how to select the optimal value of $p_i$ that leads to the optimal convergence. An adaptive strategy should be possible because we have a connection with subspace iteration. This connection may eventually shed light on how many shifts to apply per restart, in other words, how to select $p_i$ relative to $k$.

Another interesting approach is a variation of the algorithm in Table 9.1. As in that algorithm, a partial Schur decomposition is built for the wanted Ritz values that are good approximations to the desired eigenvalues $\lambda_1, \ldots, \lambda_k$. Let $j$ denote the number of locked vectors. The only difference is in the way the restart is performed. First, a set of $m - j$ shifts is determined. The shifts are the roots of a polynomial that is small on the unwanted eigenvalues $\theta_{k+1}, \ldots, \theta_{m-j}$ of $\mathbf{H}_{m-j}$. The polynomial of degree $m - j$ defined by these shifts is implicitly applied via an IRA iteration. This approach differs from the IRA iteration outlined in Table 6.1 in two ways. The first is that of deflation. The other difference is that a large-degree polynomial is applied at

every step. For symmetric **A**, Baglama, Calvetti, and Reichel [3] use Leja shifts, and this strategy outperforms `ARPACK` for small $m$. `ARPACK` uses a polynomial of degree at most $m-k$ as compared with the degree $m-j$ polynomial used by the deflated approach. Other polynomials and nonsymmetric **A** should be investigated.

**Acknowledgments.** A large part of this research was carried out while the author was writing his dissertation [16] under the kind direction of Danny Sorensen. Theorem 8.2 is a joint result with Danny that also appears in [40]. I thank Chris Beattie for showing me the matrix identity in equation (6.1). I also thank Ron Morgan for some helpful discssions on his related work [25] and Jennifer Scott for answering many questions on her codes [9, 37].

## REFERENCES

[1] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. D. Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen, *LAPACK Users' Guide*, SIAM, Philadelphia, PA, second ed., 1995.

[2] W. E. Arnoldi, *The principle of minimized iterations in the solution of the matrix eigenvalue problem*, Quart. J. Applied Mathematics, 9 (1951), pp. 17–29.

[3] J. Baglama, D. Calvetti, and L. Reichel, *Iterative methods for the computation of a few eigenvalues of a large symmetric matrix*, technical report, Department of Computer Science, Stanford University, 1995. BIT (to appear).

[4] T. Braconnier, *The Arnoldi–Tchebycheff algorithm for solving large nonsymmetric eigenproblems*, Technical Report TR/PA/93/25, CERFACS, Toulouse, France, 1993.

[5] D. Calvetti, L. Reichel, and D. C. Sorensen, *An implicitly restarted Lanczos method for large symmetric eigenvalue problems*, ETNA, 2 (1994), pp. 1–21.

[6] F. Chatelin, *Eigenvalues of Matrices*, Wiley, 1993.

[7] J. Cullum and W. E. Donath, *A block Lanczos algorithm for computing the q algebraically largest eigenvalues and a corresponding eigenspace for large, sparse symmetric matrices*, in Proceedings of the 1974 IEEE Conference on Decision and Control, New York, 1974, pp. 505–509.

[8] J. Cullum and R. A. Wilboughby, *Lanczos algorithms for large symmetric eigenvalue computations*, vol. 1, Theory, Birkhäuser, Boston, MA., 1985.

[9] I. S. Duff and J. A. Scott, *Computing selected eigenvalues of large sparse unsymmetric matrices using subspace iteration*, ACM Trans. on Math. Software, 19 (1993), pp. 137–159.

[10] G. H. Golub and C. F. V. Loan, *Matrix Computations*, Johns Hopkins, Baltimore, second ed., 1989.

[11] G. H. Golub and R. Underwood, *The block Lanczos method for computing eigenvalues*, in Mathematical Software III, J. R. Rice, ed., 1977, pp. 361–377.

[12] G. H. Golub and J. H. Wilkinson, *Ill-conditioned eigensystems and the computation of the Jordan canonical form*, SIAM Review, 18 (1976), pp. 578–619.

[13] R. G. Grimes, J. G. Lewis, and H. D. Simon, *A shifted block Lanczos algorithm for solving sparse symmetric generalized eigenproblems*, SIAM J. Matrix Analysis and Applications, 15 (1994), pp. 228–272.

[14] W. Karush, *An iterative method for finding characteristics vectors of a symmetric matrix*, Pacific J. Mathematics, 1 (1951), pp. 233–248.

[15] C. Lanczos, *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Research of the National Bureau of Standards, 45 (1950), pp. 255–282. Research Paper 2133.

[16] R. B. Lehoucq, *Analysis and Implementation of an Implicitly Restarted Iteration*, PhD thesis, Rice University, Houston, Texas, May 1995. Also available as Technical Report TR95-13, Dept. of Computational and Applied Mathematics.

[17] R. B. Lehoucq and J. A. Scott, *An evaluation of software for computing eigenvalues of sparse nonsymmetric matrices*, Preprint MCS-P547-1195, Argonne National Laboratory, Argonne, IL, 1995.

[18] R. B. Lehoucq and D. C. Sorensen, *Deflation techniques for an implicitly restarted iteration*, SIAM J. Matrix Analysis and Applications, (1996). To appear.

[19] R. B. Lehoucq, D. C. Sorensen, P. Vu, and C. Yang, *ARPACK:* An implementation of the implicitly re-started Arnoldi iteration that computes some of the eigenvalues and eigenvectors of a large sparse matrix, 1995. Available from netlib@ornl.gov under the

directory scalapack.
[20] T. A. Manteuffel, *Adaptive procedure for estimating parameters for the nonsymmetric Tchebychev iteration*, Numerische Mathematik, 31 (1978), pp. 183–208.
[21] R. S. Martin, G. Peters, and J. H. Wilkinson, *The QR algorithm for real Hessenberg matrices*, Numerische Mathematik, 14 (1978), pp. 219–231.
[22] K. Meerbergen and A. Spence, *Implicitly restarted Arnoldi with purification for the shift–invert transformation*, Technical Report TW225, Katholieke Universitet Leuven, Leuven, Belguim, April 1995. Submitted to *Mathematics of Computation*.
[23] G. S. Miminis and C. C. Paige, *Implicit shifting in the QR and related algorithms*, SIAM Journal on Matrix Analysis and Applications, 12 (1991), pp. 385–400.
[24] R. B. Morgan, *Computing interior eigenvalues of large matrices*, Linear Algebra and Its Applications, 154/156 (1991), pp. 289–309.
[25] ———, *On restarting the Arnoldi method for large nonsymmetric eigenvalue problems*, Mathematics of Computation, (1995). To appear 1996.
[26] R. B. Morgan and M. Zeng, *Estimates for interior eigenvalues of large nonsymmetric matrices*, tech. rep., 1996.
[27] C. C. Paige, *The computation of eigenvalues and eigenvectors of very large sparse matrices*, PhD thesis, University of London, London, England, 1971.
[28] C. C. Paige, B. N. Parlett, and H. A. V. der Vorst, *Approximate solutions and eigenvalue bounds from Krylov subspaces*, Numerical Linear Algebra with Applications, 2 (1995), pp. 115–134.
[29] B. N. Parlett, *Global convergence of the basic QR algorithm on Hessenberg matrices*, Mathematics of Computation, 22 (1968), pp. 803–817.
[30] ———, *The Symmetric Eigenvalue Problem*, Prentice-Hall, Englewood Cliffs, N.J., 1980.
[31] B. N. Parlett and W. G. Poole, *A geometric theory for the QR, LU, and power iterations*, SIAM J. Numerical Analysis, 10 (1973), pp. 389–412.
[32] Y. Saad, *Variations on Arnoldi's method for computing eigenelements of large unsymmetric matrices*, Linear Algebra and Its Applications, 34 (1980), pp. 269–295.
[33] ———, *Projection methods for solving large sparse eigenvalue problems*, in Matrix Pencil Proceedings, B. Kågström and A. Ruhe, eds., vol. 973 of Lecture Notes in Mathematics, Berlin, 1982, Springer–Verlag, pp. 121–144.
[34] ———, *Chebyshev acceleration techniques for solving nonsymmetric eigenvalue problems*, Mathematics of Computation, 42 (1984), pp. 567–588.
[35] ———, *Least squares polynomials in the complex plane and their use for solving sparse nonsymmetric systems*, SIAM J. Numerical Analysis, 24 (1987), pp. 155–169.
[36] ———, *Numerical Methods for Large Eigenvalue Problems*, Halsted Press, 1992.
[37] J. A. Scott, *An Arnoldi code for computing selected eigenvalues of sparse real unsymmetric matrices*, ACM Trans. Mathematical Software, 21 (1995), pp. 432–475.
[38] G. L. G. Sleijpen and H. Van der Vorst, *A Jacobi-Davidson iteration method for linear eigenvalue problems*, SIAM J. Matrix Analysis and Applications, 17, pp. 401–425.
[39] D. C. Sorensen, *Implicit application of polynomial filters in a k-step Arnoldi method*, SIAM J. Matrix Analysis and Applications, 13 (1992), pp. 357–385.
[40] ———, *Implicitly restarted Arnoldi/Lanczos methods for large scale eigenvalue calculations*, in Proceedings of the ICASE/LaRC Workshop on Parallel Numerical Algorithms, May 23-25, 1994, D. E. Keyes, A. Sameh, and V. Venkatakrishnan, eds., Norfolk, Va., 1995, Kluwer. To appear.
[41] G. W. Stewart, *Simultaneous iteration for computing invariant subspaces of non-Hermitian matrices*, Numerische Mathematik, 25 (1976), pp. 123–136.
[42] D. S. Watkins, *Understanding the QR algorithm*, SIAM Review, 24 (1982), pp. 427–439.
[43] D. S. Watkins and L. Elsner, *Convergence of algorithms of decomposition type for the eigenvalue problem*, Linear Algebra and Its Applications, 143 (1991), pp. 19–47.