ABSTRACT OF DISSERTATION

Leonard Hoffnung

The Graduate School

University of Kentucky

2004

SUBSPACE PROJECTION METHODS FOR THE QUADRATIC EIGENVALUE PROBLEM

ABSTRACT OF DISSERTATION

A dissertation submitted in partial fulfillment of the
requirements of the degree of Doctor of Philosophy in the
College of Arts and Sciences
at the University of Kentucky

By

Leonard Hoffnung

Lexington, Kentucky

Co-Directors: Dr. Ren-Cang Li, Department of Mathematics
and Dr. Qiang Ye, Department of Mathematics

Lexington, Kentucky

2004

ABSTRACT OF DISSERTATION

SUBSPACE PROJECTION METHODS FOR THE QUADRATIC EIGENVALUE PROBLEM

Model reduction of the quadratic eigenvalue problem is an area of considerable recent interest. Such eigenvalue problems arise in control theory applications, acoustics, and structural analysis, where the dominant behavior of the system is determined by a relatively small number of eigenvalues. As such models can be very large, directly computing all eigenpairs is impractical. Instead, the system is reduced to a matrix equation of much smaller degree which is computationally amenable. Customarily, this is done through a linearization procedure.

In this thesis, we present model reduction techniques that construct a reduced-order model which is also given by a quadratic eigenvalue problem. In chapter 2, we describe a Krylov-type projection method that reduces a symmetric monic quadratic eigenvalue problem to another symmetric QEP of banded structure. We also describe a Rayleigh-Ritz procedure for subspace enlargement which accelerates convergence of the projected problem to a desired eigenpair. In chapter 3, we examine several linearization techniques and their expected rates of convergence using a moment-matching result of Grimme. In addition, we develop a variant of nonsymmetric Lanczos that reduces a monic QEP to one of triangular-Hessenberg form, with optimal orders of moment-matching.

Keywords: quadratic eigenvalue problem, structure-preserving model reduction, linearization, moment matching, Krylov methods.

(Leonard Hoffnung)

(Date)

SUBSPACE PROJECTION METHODS FOR THE QUADRATIC EIGENVALUE PROBLEM

By

Leonard Hoffnung

Co-Director of Dissertation

Co-Director of Dissertation

Director of Graduate Studies

(Date)

RULES FOR THE USE OF DISSERTATIONS

Unpublished dissertations submitted for the Doctor's degree and deposited in the University of Kentucky Library are as a rule open for inspection, but are to be used only with due regard to the rights of the authors. Bibliographical references may be noted, but quotations or summaries of parts may be published only with the permission of the author, and with the usual scholarly acknowledgments.

Extensive copying or publication of the dissertation in whole or in part also requires the consent of the Dean of the Graduate School of the University of Kentucky.

A library that borrows this dissertation for use by its patrons is expected to secure the signature of each user.

Name and Address Date

DISSERTATION

Leonard Hoffnung

The Graduate School

University of Kentucky

2004

SUBSPACE PROJECTION METHODS FOR THE QUADRATIC EIGENVALUE PROBLEM

DISSERTATION

A dissertation submitted in partial fulfillment of the
requirements of the degree of Doctor of Philosophy in the
College of Arts and Sciences
at the University of Kentucky

By

Leonard Hoffnung

Lexington, Kentucky

Co-Directors: Dr. Ren-Cang Li, Department of Mathematics
and Dr. Qiang Ye, Department of Mathematics

Lexington, Kentucky

2004

ACKNOWLEDGMENTS

I would like to thank the members of my advisory committee for their assistance and help throughout my graduate education. I'd especially like to thank my advisors, Ren-Cang Li and Qiang Ye, for their encouragement and advice. Thanks also to Zhaojun Bai for his thoughts and research questions, one of which supplies a main theme motivating this thesis. Additionally, thanks to the mathematicians from outside the University of Kentucky who have influenced my mathematical development, including my undergraduate advisor David Kammler of Southern Illinois University and Jane Cullum of Los Alamos National Laboratory. Most of all, thanks to my wife Sveta for her love, support, and patience.

Contents

List of Figures

Chapter 1

Preliminaries

1.1 Introduction

Quadratic eigenvalue problems arise in many engineering fields, such as acoustics, structural analysis, and control theory. In many applications, the behavior of the physical phenomenon in question can be described by a second-order differential equation. By using a discretization technique, the continuous form of this differential equation can be approximated by a second-order discrete (matrix) formulation. The numerical solution is then a vector-valued function of time $y(t)$, satisfying

$$Ay'' + By' + C = f(t)$$

$$y(0) = g$$

$$y'(0) = h$$

where $f(t)$ is a time-dependent input and g, h are the initial conditions at time $t = 0$. Here, A, B, and C are matrices obtained from the discretization process. Separating $y(t) = e^{\lambda t}x$ gives the customary formulation of the quadratic eigenvalue problem: find a constant vector $x \neq 0$ and a scalar λ so that

$$(\lambda^2 A + \lambda B + C)x = 0. \tag{1.1}$$

Here, λ is called an *eigenvalue*, and x is its corresponding *eigenvector*.

The behavior of the solution over time is apparent from its eigenvalue; oscillatory behavior is determined by the imaginary part, and exponential growth/decay of the solution is determined by the real part. Exponential growth is undesirable; in a structure such as a bridge, this indicates a dangerous resonant frequency that could even result in the structure's total collapse under the right conditions. In a model of an aircraft, the pilot could lose control altogether during turbulence. Therefore, the eigenvalues with largest real part are of primary importance; by computing such eigenvalues, dangerous flaws in an engineering project can be found and corrected during the design stage.

The challenge is to solve a given quadratic eigenvalue problem as accurately and efficiently as possible. Problems of interest in industry tend to be quite large, currently of order ten million or more. The QR method for the linear eigenvalue problem $Ax = \lambda x$ and the related QZ method for the generalized eigenvalue problem $Ax = \lambda Bx$ use matrix operations to transform the eigenvalue problem to a simple form, from which the eigenvalues are easily found. These methods, called *direct methods* because they almost always converge within a given number of iterations (determined by the desired tolerance and matrix size), are reasonably robust, well understood, and reliable. Unfortunately, direct methods are too expensive to be practical for problems of this scale. Instead, we construct another eigenvalue problem which approximates the original, but has much smaller order; this idea is called *model reduction.* This reduced-order problem then can be solved with direct methods at a modest cost.

Traditionally, a quadratic eigenvalue problem is handled by constructing an equivalent linear eigenvalue problem of twice the dimension, a process called *linearization.*

There are a number of ways to do this, for example

$$\begin{pmatrix} 0 & I \\ -C & -B \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \lambda \begin{pmatrix} x \\ y \end{pmatrix}. \tag{1.2}$$

If λ, x, y satisfy Equation (1.2), then λ, x also satisfy Equation (1.1). Thus, a linearization of a quadratic eigenvalue problem has the same spectrum as the original.

Once a linearization has been constructed, we may construct a reduced-order model of the linearization which is solved by a direct method. There are some disadvantages to introducing the additional linearization step, however. Doubling the dimension means that the corresponding storage requirement for every vector is also doubled. The quadratic eigenvalue problem often has special structure arising from its application, e.g. in structural mechanics, A could be symmetric positive definite while B and C and symmetric positive semi-definite; the linearized problem may fail to preserve the special structure. This raises the following question: can we find efficient model reduction techniques which operate directly on a quadratic eigenvalue problem to yield another, smaller rank quadratic problem? In this work, we present several approaches for model reduction which attempt to answer this question.

One model reduction technique discussed in this thesis is the idea of subspace projection. In a projection method, an orthonormal basis Q_m is constructed by a suitable means, and Equation (1.1) is approximated by the quadratic eigenvalue problem $Q_m^T(\mu^2 A + \mu B + C)Q_m u = 0$. This reduced-order problem has *Ritz values* μ and *Ritz vectors* $Q_m u$ which approximate the eigenvalues and eigenvectors, respectively, of Equation (1.1). An appealing feature of projection methods is that the symmetry or skew-symmetry of A, B, C is preserved in the model reduction, as are the spectral characteristics possessed by quadratic eigenvalue problems with such structure (for example, the eigenvalues of symmetric quadratic eigenvalue problems occur in conjugate pairs). Projection techniques are discussed in Chapters 2-3.

The organization of the thesis is as follows. Chapter 1 is dedicated to background material. Section 1.2 covers necessary reference material from numerical linear algebra, including symmetric matrix factorizations and a review of Krylov methods for linear problems. Section 1.3 gives an overview of Raviart-Thomas basis elements and the finite element method, a discretization technique commonly used for constructing a discrete model of a physical problem domain (and used to generate the test problems from our numerical examples).

Chapter 2 begins by describing current research into the quadratic eigenvalue problem, in particular the linearization-based algorithms by Tisseur and Mehrmann et al. The remainder of the chapter describes new projection methods for model reduction of the quadratic eigenvalue problem. In Section 2.2, we present our recent work on a Krylov-type subspace projection method operating directly on a monic quadratic eigenvalue problem (with $A = I$) rather than a linearization. As in the case of the Lanczos or Arnoldi methods applied to a linear eigenvalue problem, the resulting reduced-order eigenvalue problem has a special structure. For the symmetric quadratic eigenvalue problem, a Lanczos-type method is presented which produces a reduced-order symmetric eigenvalue problem where each matrix has a banded structure. In the nonsymmetric case, analogous Arnoldi-type and nonsymmetric Lanczos-type algorithms are constructed which produce reduced-order eigenvalue problems with a Hessenberg-like or banded structure, respectively.

The Krylov-type methods in Section 2.2 determine each vector of Q_m by the type of recurrence and the desired structure of the reduced-order problem. In Section 2.3, we present three projection methods that use different applications of the Arnoldi method to enlarge the basis Q_m. These methods offer greater flexibility in the construction of the projected subspace, although some of the structure in the reduced-order problem is lost. A discussion of spectral transformation for the quadratic eigenvalue problem appears in Section 2.4.

Section 2.5 generalizes the methods from Sections 2.2, 2.3 and presents a unified projection algorithm. In each iteration, the subspace spanned by Q_m is enlarged using the following selection criterion: choose q_{m+1} so that the "best" solution to the order m reduced eigenvalue problem has as large a residual as possible when substituted into the subsequent reduced-order eigenvalue problem. Here, "best" can be defined in various ways; the natural choice of a Ritz pair is demonstrated to give good results.

Chapter 3 gives a moment-matching result of Villemagne and Skelton for the generalized eigenvalue problem, and applies it to a linearization of the QEP. The methods in this chapter are model reductions that attempt to match as many as possible of the moments of the original eigenvalue problem. Bai's SOAR algorithm (Section 3.2.1) and the Q-Arnoldi algorithm of Meerbergen and Robbé (Section 3.2.2) are projection methods which use the Arnoldi method to obtain m matching moments. Section 3.2.3 gives a modification of SOAR for the symmetric quadratic eigenvalue problem that offers comparable moment-matching and reduced storage costs. Section 3.3 presents a model reduction method based on nonsymmetric Lanczos; while not a projection method, the resulting reduced-order problem preserves a triangular-Hessenberg structure, and matches the optimal number of $4m$ moments. Lastly, Chapter 4 contains numerical examples comparing and contrasting the behavior of the algorithms on a sample problem in dissipative acoustics. The best results are obtained with the method suggested in Section 2.5. Among moment-matching techniques, the fastest eigenvalue convergence is obtained with the triangular-Hessenberg model reduction from Section 3.3.

1.2 Review of numerical linear algebra

The background material in this section can be found in any standard numerical linear algebra text [22, 16, 24]; see also Parlett [41], Cullum and Willoughby [11].

1.2.1 Cholesky factorization

A symmetric matrix A has a Cholesky factorization if it can be written as the product $A = LL^T$, where L is a nonsingular lower triangular matrix (called the Cholesky factor of A). It is easy to see that if A has a Cholesky factorization, then A must be symmetric positive definite; in fact, the converse is also true. This provides a convenient and numerically reliable characterization of symmetric positive definite matrices.

Theorem 1.2.1. *If A is a symmetric positive definite matrix, then there exists a unique lower triangular L with positive diagonal elements so that $A = LL^T$.*

Proof. The proof is by induction. Clearly the result holds for $n = 1$. Suppose the result holds for k. Write the symmetric positive definite matrix A_{k+1} as

$$A_{k+1} = \begin{pmatrix} A_k & b \\ b^T & c \end{pmatrix}.$$

By induction, $A_k = L_k L_k^T$ has a unique Cholesky factorization. Let $d = L_k^{-1} b$ and $v = \begin{pmatrix} L_k^{-T} d \\ -1 \end{pmatrix}$. Since v is nonzero, the quantity $s = v^T A_{k+1} v = c - d^T d$ must be positive. Then it is easy to check that

$$\begin{aligned} \begin{pmatrix} L_k & \\ d^T & \sqrt{s} \end{pmatrix} \begin{pmatrix} L_k^T & d \\ & \sqrt{s} \end{pmatrix} &= \begin{pmatrix} A_k & b \\ b^T & s + d^T d \end{pmatrix} \\ &= A_{k+1}. \end{aligned}$$

Note that with the requirement that the Cholesky factor has only positive elements on the diagonal, the factorization is uniquely determined. □

Furthermore, we have a constructive technique for computing the Cholesky factorization. After step k, we have computed the first k rows of L. Then, we compute the $(k+1)$-st row of L by setting $j = 1, 2, \ldots, k+1$, and solving the identity

$$a_{k+1,j} = \sum_{m=1}^{j} l_{k+1,m} l_{jm} \tag{1.3}$$

for the element $l_{k+1,j}$.

Additionally, this provides a simple test for positive definiteness: if we attempt to solve Equation (1.3) for the diagonal element $l_{k+1,k+1}$, only to find that

$$l_{k+1,k+1}^2 = a_{k+1,k+1} - \sum_{m=1}^{k} l_{k+1,m}^2$$

is negative, then the matrix A is not positive definite. This test is not too expensive, and is numerically reliable [24, Section 10.1].

Fill-in of the Cholesky factor

In practice, we would want a computed Cholesky factor of a sparse matrix to remain sparse, if at all possible. For banded matrices of low bandwidth, the Cholesky factor does remain small: if $A = LL^T$ has bandwidth $2d + 1$ (i.e. $a_{ij} = 0$ for all $|i - j| > d$),

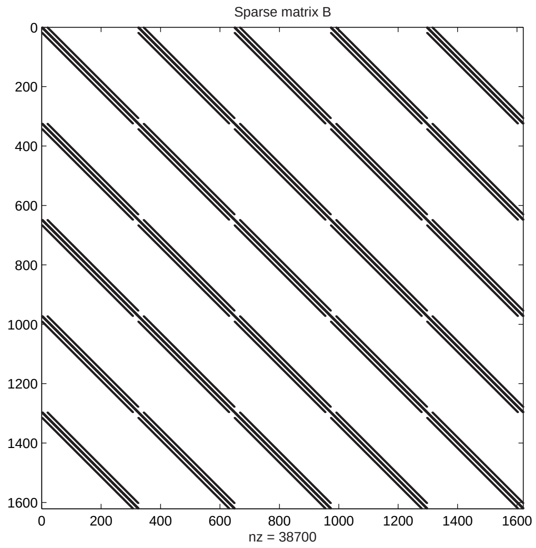


Figure 1.1: Sparse matrix B (sparsity approx. 1.5%)

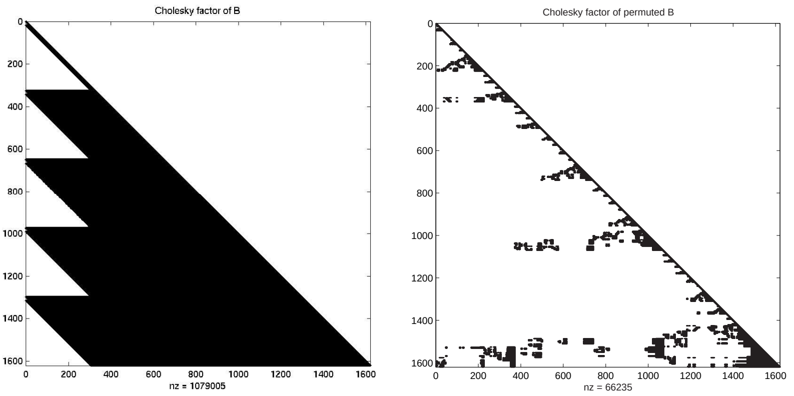
then L must have bandwidth at most $d+1$. Note, however, that if the bandwidth of A is large then the Cholesky factor may be dense, even if A is very sparse. Consider the following example. Let A be the five-point finite difference discretization of the Laplacian on an 18×18 square mesh (an order 324 sparse matrix). Construct a symmetric positive definite B by tiling A in a 5×5 array, and adding an identity:

$$B = \begin{pmatrix} A & A & A & A & A \\ A & A & A & A & A \\ A & A & A & A & A \\ A & A & A & A & A \\ A & A & A & A & A \end{pmatrix} + I.$$

This matrix is depicted in Figure 1.1.

Although B is sparse (containing less than 1.5% nonzeros), computing its factorization directly produces a very dense Cholesky factor consisting of 82% nonzero elements (Figure 1.2(a)). Such a matrix would be costly to compute and store in a large-scale computation.

This difficulty can often be remedied by choosing a suitable permutation P so that the nonzero elements of P^TAP are closer to the diagonal, and utilizing the Cholesky



(a) Factor of unpermuted B (b) Factor of permuted B

Figure 1.2: Cholesky factors, with and without permutation

decomposition of the permuted matrix P^TAP instead. Such Cholesky factors are generally no denser than A^TA. Efficient algorithms for choosing such a permutation have been devised by Tim Davis [1] and others.

The difference in sparsity can be dramatic; constructing a permutation of B using the `symmmd` function from Matlab and factoring $P^TBP = LL^T$ produces the Cholesky factor containing 5% nonzeros shown in Figure 1.2(b).

1.2.2 LDL^T factorization of indefinite matrices

A natural idea would be to try to find an extension of the Cholesky factorization to symmetric but indefinite matrices. One might look for a factorization of the form $A = LDL^T$, where D is a diagonal matrix and L is lower triangular. Unfortunately, such a factorization need not exist, or might be highly unstable. As an example, consider the matrix $\begin{pmatrix} \epsilon & 1 \\ 1 & \epsilon \end{pmatrix}$; for small $\epsilon > 0$, we get ill-behaved factors

$$D = \begin{pmatrix} \epsilon & 0 \\ 0 & \epsilon - 1/\epsilon \end{pmatrix}, \qquad L = \begin{pmatrix} 1 & 1/\epsilon \\ 0 & 1 \end{pmatrix},$$

while the factorization does not exist at all for $\epsilon = 0$. Observe also that applying a symmetric permutation to this matrix leaves it unchanged; therefore, pivoting alone is not sufficient to fix the situation. However, if we allow D to be block diagonal with 1×1 or 2×2 blocks, then a permutation P can be found so that the factorization $P^TAP = LDL^T$ exists. Without loss of generality, we can scale D, L so that L is unit triangular (for convenience).

Proof. The following argument is called the *diagonal pivoting method* in the book by Higham [24, Section 10.4]. Choose a permutation P so that $P^TAP = \begin{pmatrix} D & C^T \\ C & B \end{pmatrix}$, where D is a 1×1 or 2×2 nonsingular block. Such a permutation must exist unless A is identically zero (a trivial case). Then we can factor

$$P^TAP = \begin{pmatrix} I & 0 \\ CD^{-1} & I \end{pmatrix} \begin{pmatrix} D & 0 \\ 0 & B - CD^{-1}C^T \end{pmatrix} \begin{pmatrix} I & D^{-1}C^T \\ 0 & I \end{pmatrix}.$$

The submatrix $B - CD^{-1}C^T$ can be permuted and factored similarly. This process is repeated until A is factored completely. □

For numerical stability, however, the precise choice of P is important. Higham [24] discusses a complete pivoting strategy by Bunch and Parlett, and an $O(n^2)$ partial pivoting strategy by Bunch and Kaufman. It turns out that while complete pivoting guarantees stability, the cheaper partial pivoting strategy also works well in practice.

1.2.3 Krylov methods

Krylov methods are a class of algorithms which apply the Rayleigh-Ritz procedure to a particular natural choice of subspace (the Krylov subspace). Recall that the Rayleigh-Ritz procedure approximates a large real matrix A by projecting it onto a given subspace $\mathcal{S}$ of relatively small dimension m. Letting Q_m be an orthonormal basis of $\mathcal{S}$, we can construct the $m \times m$ matrix $A_m = Q_m^T A Q_m$. The eigenvalues $\theta_1, \ldots, \theta_m$ of A_m, also called Ritz values, approximate some of the eigenvalues of A. In the case of Krylov methods, the subspace used for the projection is the *Krylov subspace* of dimension m

$$\mathcal{K}_m(A, q) = \text{span}\{q, Aq, \ldots, A^{m-1}q\}$$

where $q \neq 0$ is an initial vector, often randomly chosen. Note that the above basis is never used numerically since it is poorly conditioned; $A^m q$ approaches a dominant eigenvector as m increases, so successive vectors $A^m q$ and $A^{m+1}q$ will be nearly parallel. Instead, practical Krylov-based algorithms produce bases which are equivalent in exact arithmetic and better conditioned.

The Arnoldi method

The simplest Krylov method for a nonsymmetric matrix A is the Arnoldi method. For $i = 1, 2, \ldots, m-1$, we inductively produce an orthonormal basis $\{q_1, \ldots, q_i\}$ of $\mathcal{K}_{i+1}(A, q)$ from the previous basis of $\mathcal{K}_i(A, q)$. Clearly $q_1 = q/\|q\|$ is a basis of $\mathcal{K}_1(A, q)$. For each i, we have $\mathcal{K}_{i+1} = \text{span}\{\mathcal{K}_i, Aq_i\}$; to extend $\{q_1, \ldots, q_i\}$ to a basis of $\mathcal{K}_{i+1}$, we need to compute the product $r = Aq_i$, project off its components in $\mathcal{K}_i$ using the Gram-Schmidt process, and set q_{i+1} equal to the resulting unit vector. In matrix form, after m steps we have

$$AQ_m = Q_m H_m + h_{m+1,m} e_m q_{m+1}^T \tag{1.4}$$

where H_m is an upper Hessenberg matrix whose entries were determined from the Gram-Schmidt reorthogonalization at each step i. H_m is clearly the desired projection of A onto the Krylov subspace of dimension m; computing the eigenvalues of H_m by a direct method (i.e. QR iteration) is not too expensive as long as m is small.

There are several disadvantages to the Arnoldi method when m is of even moderate size. All of the previous basis vectors $q_1, q_2, \ldots, q_m$ must be kept available at each step of the algorithm, driving up storage costs. The $O(m^3)$ cost of computing the Ritz values can become large. Nonetheless, the Arnoldi method is useful in many applications, especially with restarting. One widely used and publicly-available implementation appears in ARPACK [32].

The Lanczos method

In the special case when A is symmetric, the Arnoldi method reduces to a simplified algorithm known as the Lanczos method. Instead of a Hessenberg matrix H_m, a symmetric tridiagonal matrix T_m is obtained. This is clear intuitively; using the corresponding Arnoldi relation

$$AQ_m = Q_m T_m + t_{m+1,m} e_m q_{m+1}^T \tag{1.5}$$

it follows that $T_m = Q_m^T A Q_m$ is both symmetric and Hessenberg, and therefore tridiagonal. Then the construction in Equation (1.5) can be written as the following three-term recurrence. Letting $q_1 = q/\|q\|$ be an initial unit vector with $\alpha_1 = q_1^T A q_1$ and $\beta_1 = 0$, we have for $j = 1, \ldots, m-1$,

$$\begin{aligned} \beta_{j+1} q_{j+1} &= A q_j - \alpha_j q_j - \beta_j q_{j-1} \\ \alpha_{j+1} &= q_{j+1}^T A q_{j+1} \end{aligned} \tag{1.6}$$

where $\{\alpha_1, \ldots\}$ and $\{\beta_2, \ldots\}$ are the diagonal and off-diagonal entries of T_m

$$T_m = \begin{pmatrix} \alpha_1 & \beta_2 & & \\ \beta_2 & \alpha_2 & \ddots & \\ & \ddots & \ddots & \beta_m \\ & & \beta_m & \alpha_m \end{pmatrix}. \tag{1.7}$$

More formally, it is not difficult to use the defining recurrence in (1.6) to prove that the generated basis Q_m is orthonormal and Equation (1.5) holds.

The Lanczos method offers significant improvements in speed and efficiency over the Arnoldi method, and is well suited to large, sparse problems. Memory usage is minimal; at step m we require storage for the $2m+1$ distinct nonzero elements of T_{m+1} and the three Lanczos vectors q_{m-1}, q_m, q_{m+1}. Furthermore, the eigenvalues and eigenvectors of the tridiagonal T_m can be computed cheaply by means of Sturm sequencing and inverse iteration [12], respectively.

Eigenpairs of a symmetric tridiagonal matrix matrix

In this section, we consider the problem of computing the eigenpairs of the symmetric tridiagonal matrix T_m. First, we will show how to find the eigenvalues using bisection. Suppose that T_{k+1} has the matrix T_k as its principal minor, and neither matrix is singular. Recall that determinants are invariant under similarity transformations; in particular, the determinant of a symmetric matrix $T = Q^T \Lambda Q$ is simply the product of its eigenvalues. Therefore, if T_k has j negative eigenvalues then $\text{sign}(\det(T_k)) = (-1)^j$. By the interlacing property, T_{k+1} must have either j or $j+1$ negative eigenvalues. Thus, either the signs of $\det(T_k), \det(T_{k+1})$ match and T_k, T_{k+1} have the same number of negative eigenvalues, or the signs differ and T_{k+1} has one additional negative eigenvalue.

If the determinants of a nested series $T_1, \ldots, T_m$ of principal minors are available, then counting the number of times consecutive determinants change sign gives the number of negative eigenvalues of T_m. Replacing T_m with the shifted matrix $(T_m - \sigma I_m)$ gives the following statement:

Definition 1.2.2 (Sturm Sequencing). Let $T_1, T_2, \ldots, T_m$ be the principal minors of a symmetric matrix T_m. For a fixed σ, construct the sequence $\{a_i(\sigma)\}$, $i = 0, 1, \ldots, m$ so that $a_0 = 1$ by definition and $a_i(\sigma) = \det(T_i - \sigma)$ for $i > 0$. The number of times consecutive terms in this sequence differ in sign equals the quantity of eigenvalues of T_m which are less than σ.

In general, determinants can be expensive to compute and numerically unstable, which would appear to be a problem. For our purpose, however, it is sufficient to compute the ratios of consecutive determinants, which are less prone to over- or underflow. For tridiagonal T_m labelled as in Equation (1.7), the determinants satisfy

$$\det(T_{k+1}) = \alpha_{k+1} \det(T_k) - \beta_{k+1}^2 \det(T_{k-1}).$$

Dividing by $\det(T_k)$ gives an identity in terms of ratios:

$$\left[\frac{\det(T_{k+1})}{\det(T_k)}\right] = \alpha_{k+1} - \beta_{k+1}^2 \left[\frac{\det(T_{k-1})}{\det(T_k)}\right].$$

Replacing T_k by the shifted matrix $T_k - \sigma I_k$ gives a stable recurrence for the ratios of consecutive determinants $r_i(\sigma) = a_i(\sigma)/a_{i-1}(\sigma)$:

$$r_i(\sigma) = \begin{cases} (\alpha_1 - \sigma), & i = 1, \\ (\alpha_i - \sigma) - \beta_i^2 / r_{i-1}(\sigma), & i = 2, \ldots, m. \end{cases} \tag{1.8}$$

Thus, the number of eigenvalues of T_m which are smaller than σ can be determined in $O(m)$ time using (1.8) by counting how many negative values appear in the set

$\{r_1(\sigma), r_2(\sigma), \ldots, r_m(\sigma)\}$. A bisection procedure (used, for example, in EISPACK [50]) can then be used to isolate each eigenvalue within a desired tolerance. Note that this parallelizes well, since after an initial rough subdivision of the spectrum into intervals, each processor can find all of the eigenvalues in its interval without additional communication (see also [15]).

Once an approximate eigenvalue $\mu_k \approx \lambda_k$ of T has been computed, inverse iteration is an effective method for finding the corresponding eigenvector x_k (provided that λ_k is sufficiently well separated from other eigenvalues). The idea is that if the vector v_i is not orthogonal to x_k, then solving $(T - \mu_k)v_{i+1} = v_i$ produces a vector v_{i+1} in which the x_k component is magnified. More precisely, if $v_0 = \sum_{i=1}^{m} \alpha_i x_i$, then

$$\begin{aligned} v_1 &= (T - \mu_k)^{-1} v_0 \\ &= \sum_{i=1}^{n} \left(\frac{\alpha_i}{\lambda_i - \mu_k} \right) x_i \end{aligned} \tag{1.9}$$

and after j iterations we obtain

$$v_j = \sum_{i=1}^{n} \left(\frac{\alpha_i}{(\lambda_i - \mu_k)^j} \right) x_i. \tag{1.10}$$

The angle between v_j and the desired eigenvector is bounded by

$$\begin{aligned} \cos^2\langle x_k, v_j \rangle &= \left(\frac{\alpha_k}{(\lambda_k - \mu_k)^j} \right)^2 \Big/ \sum_{i=1}^{n} \frac{\alpha_i^2}{(\lambda_i - \mu_k)^{2j}} \\ &\geq \frac{\alpha_k^2}{\sum_{i \neq k} \alpha_i^2 \delta^{2j} + \alpha_k^2} \\ &= \frac{1}{1 + \delta^{2j} \tan^2\langle v_0, x_k \rangle} \end{aligned} \tag{1.11}$$

where $\delta = \dfrac{|\lambda_k - \mu_k|}{\min_{i \neq k} |\lambda_i - \mu_k|}$ is the relative gap between μ_k and the second-closest eigenvalue of T. Thus, convergence is guaranteed (in exact arithmetic) if $\delta < 1$ and v_0 is not orthogonal to x_k. In practice, we often have $\delta \ll 1$ when λ_k is distinct, and almost any randomly-chosen initial vector v_0 will be adequate. Therefore, convergence generally occurs after only a few iterations.

The last technical question is how to perform the linear solves economically. Cullum [11, 12] uses the QR factorization of $(T - \mu_k I)$. For our special case with T tridiagonal, the triangular matrix R has bandwidth three and the factorization can be computed in $O(m)$ time (e.g. by zeroing the subdiagonal entries with Givens rotations). Once the factorization is computed, each solve is performed in $O(m)$ time by solving $Rv_{j+1} = Q^T v_j$ by back substitution. Thus, this method is economical even for large tridiagonal matrices.

Note that inverse iteration still converges even though $T - \mu_k$ is near singular. Indeed, the error in computing v_j tends to be quite large, but most of this error is in the direction of the eigenvector x_k and therefore doesn't affect convergence adversely [41]. In the case of clustered eigenvalues, inverse iteration may not distinguish the eigenvectors sufficiently accurately. Work by Dhillon [43, 17] uses careful shifting and twisted factorization to compute eigenvalues and eigenvectors to high relative accuracy.

Theorem 1.2.3. *Suppose a real symmetric $n \times n$ matrix A has eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$ with corresponding orthonormal eigenvectors $x_1, x_2, \ldots, x_n$. Let T_k be obtained from k steps of Lanczos method with the initial unit vector q_1 (assuming no breakdown occurs). Then the gap between the largest Ritz value θ_1 of T_k and λ_1 is bounded by*

$$0 \leq \lambda_1 - \theta_1 \leq (\lambda_1 - \lambda_n) \tan^2\langle q_1, x_1 \rangle \min_{\substack{p \in \mathcal{P}_k \\ p(\lambda_1)=1}} \max_{i=2,3,\ldots,n} |p(\lambda_i)|^2.$$

Proof. We know that $\lambda_1 - \theta_1 \geq 0$ using the Cauchy criterion. Then

$$\begin{aligned}
\lambda_1 - \theta_1 &= \lambda_1 - \max_{\substack{x \in \mathcal{K}_k \\ x \neq 0}} \frac{x^T A x}{x^T x} \\
&= \lambda_1 - \max_{\substack{p \in \mathcal{P}_k \\ p(A) q_1 \neq 0}} \frac{q_1^T p(A)^2 A q_1}{q_1^T p(A)^2 q_1},
\end{aligned}$$

writing $x \in \mathcal{K}_k(A, q_1)$ as $p(A)q_1$ for some polynomial p of degree k. Now expressing $q_1 = \sum_{k=1}^{n} \alpha_i x_i$ as a linear combination of the eigenvectors of A, we have

$$\begin{aligned}
\lambda_1 - \theta_1 &= \lambda_1 - \max_{\substack{p \in \mathcal{P}_k \\ p(A) q_1 \neq 0}} \frac{\sum_{i=1}^{n} \alpha_i^2 p(\lambda_i)^2 \lambda_i}{\sum_{i=1}^{n} \alpha_i^2 p(\lambda_i)^2} \\
&= \min_{\substack{p \in \mathcal{P}_k \\ p(A) q_1 \neq 0}} \frac{\sum_{i=1}^{n} \alpha_i^2 p(\lambda_i)^2 (\lambda_i - \lambda_1)}{\sum_{i=1}^{n} \alpha_i^2 p(\lambda_i)^2} \\
&\leq (\lambda_1 - \lambda_n) \min_{\substack{p \in \mathcal{P}_k \\ p(A) q_1 \neq 0}} \frac{\sum_{i=2}^{n} \alpha_i^2 p(\lambda_i)^2}{\sum_{i=1}^{n} \alpha_i^2 p(\lambda_i)^2} \\
&\leq (\lambda_1 - \lambda_n) \min_{\substack{p \in \mathcal{P}_k \\ p(A) q_1 \neq 0}} \frac{(1 - \alpha_1^2) \max_{i=2,\ldots,n} p(\lambda_i)^2}{\sum_{i=1}^{n} \alpha_i^2 p(\lambda_i)^2}
\end{aligned}$$

Lastly, with the assumptions that $p(\lambda_i) \neq 0$ for an optimal choice of polynomial p, and the initial vector q_1 is not orthogonal to x_1, we can bound the denominator below by $\alpha_1^2 p(\lambda_1)^2$ to get

$$\begin{aligned}
\lambda_1 - \theta_1 &\leq (\lambda_1 - \lambda_n) \frac{1 - \alpha_1^2}{\alpha_1^2} \min_{\substack{p \in \mathcal{P}_k \\ p(A) q_1 \neq 0}} \left(\frac{\max_{i=2,\ldots,n} p(\lambda_i)^2}{p(\lambda_1)^2} \right) \\
&= (\lambda_1 - \lambda_n) \tan^2 \langle q_1, x_1 \rangle \min_{\substack{p \in \mathcal{P}_k \\ p(\lambda_1) = 1}} \max_{i=2,3,\ldots,n} |p(\lambda_i)|^2.
\end{aligned}$$

□

Thus, the largest Ritz value approximates the largest eigenvalue well if there exists a degree k polynomial p which equals 1 at λ_1 and is small at all the other eigenvalues of A. This is the case when λ_1 is well-separated from the other eigenvalues. However, if λ_1 is tightly clustered, then convergence can be very slow. This is in fact the case in practice: well-separated extreme eigenvalues tend to converge very quickly, while clustered and/or interior eigenvalues converge more slowly (if at all). If a small number of tightly-clustered extreme eigenvalues are at the same time well-separated from the rest of the spectrum, block Lanczos methods remedy this problem somewhat at the expense of higher memory usage; these methods will be discussed later.

To obtain a more explicit bound, the inequality in Theorem 1.2.3 can be relaxed further by replacing the minimax expression with a Chebyshev polynomial. A similar argument can be used to bound the gaps between the smallest eigenvalue and Ritz value; a more complicated expression determining the convergence of interior Ritz values appears in Saad [49].

Other characteristic properties of the Lanczos method include:

- Shifting and scaling the original matrix does not affect the convergence of the Ritz values (this is easy to check). Accelerating the convergence of interior, clustered Ritz values is done by a spectral transformation (as in shift-and-invert Lanczos).

- Convergence to the desired eigenpair λ_1, x_1 may not occur if the initial vector q_1 is orthogonal or nearly orthogonal to x_1. Similarly, multiplicity of eigenvalues cannot be determined reliably with single-vector Lanczos.

- While breakdown can occur (the succeeding off-diagonal element β_{m+1} may be zero), this is beneficial; such a breakdown indicates that an invariant subspace has been found and all Ritz pairs have converged.

Variations on Lanczos method

Several variations on the original symmetric Lanczos method are worth mentioning at this point. These include nonsymmetric Lanczos methods, generalized symmetric Lanczos, and block Lanczos.

The nonsymmetric Lanczos method is an iterative method that produces a tridiagonal projection of a nonsymmetric matrix A using the Krylov spaces of both A and its transpose. Start with vectors u_1, v_1 satisfying $u_1^T v_1 = 1$. It is generally possible to extend u_1, v_1 to biorthogonal U_m, V_m (meaning that $U_m^T V_m = I$) and the following recurrences are satisfied:

$$AV_m = V_m T_m + \beta_{m+1} v_{m+1} e_m^T \tag{1.12}$$

$$A^T U_m = U_m T_m^T + \gamma_{m+1} u_{m+1} e_m^T. \tag{1.13}$$

The columns of U_m, V_m thus generated form bases of the left and right Krylov spaces $\mathcal{K}_m(A^T, u_1)$, $\mathcal{K}_m(A, v_1)$ respectively. This fact will be used later in section 3.3.

There are a number of differences in behavior between the symmetric and nonsymmetric Lanczos behavior. One can observe that the biorthogonality constraint and Equations (1.12)–(1.13) determine U_m, V_m only up to scale; in most implementations, this ambiguity is resolved by scaling the Lanczos vectors so that $0 \leq \beta_i = \pm\gamma_i$ for each i. Also, there are now several ways in which the recurrence can fail. If one or both of the vectors u_{m+1}, v_{m+1} are very small, then the near-zero vector(s) can be replaced by a new randomly-chosen vector (suitably orthogonalized). If u_{m+1}, v_{m+1} are nearly orthogonal, then the algorithm as described must terminate. Lookahead [45] and new-start [57] are approaches to correct for this breakdown. Lookahead temporarily enlarges the block size so that the pivot element is replaced by a nonsingular block. New-start corrects the breakdown by adding a new Lanczos vector and increasing the length of the recurrences (1.12), (1.13). In both cases, the projection T_m is no longer strictly tridiagonal; instead, it acquires a bulge or increases bandwidth.

Another modification of the Lanczos algorithm, the symmetric indefinite Lanczos method [42, 31], approximates a symmetric indefinite matrix pencil $Ax = \lambda Bx$ by a symmetric tridiagonal equation $T_m u = \mu D_m u$. Note that this algorithm isn't used for pencils where A and/or B are definite; in that case, the implicit application of a Cholesky factorization to the definite matrix reduces the problem to the usual symmetric eigenvalue problem. This process generates a basis W_m of the Krylov space $\mathcal{K}_m(B^{-1}A, w_1)$ so that $W_m^T A W_m = T_m$ and $W_m^T B W_m = D_m$ is diagonal (but not necessarily the identity). The appropriate recurrence is

$$B^{-1}AW_m = W_m D_m^{-1} T_m + (\beta_{m+1}/d_{m+1}) w_{m+1} e_m^T. \tag{1.14}$$

This method can be viewed as a special case of nonsymmetric Lanczos; observe that $U_m = BW_m D_m^{-1}$, $V_m = W_m$ form a biorthogonal basis which tridiagonalizes $B^{-1}A$. Therefore, symmetric indefinite Lanczos inherits the features of the nonsymmetric Lanczos algorithm, such as breakdown (when some $d_i = 0$). See section 3.2.3 for an application of nonsymmetric Lanczos to a linearized quadratic eigenvalue problem.

Lastly, the block Lanczos method replaces each Lanczos vector q_i in the symmetric Lanczos method with a block Q_i of orthonormal vectors. Starting with an initial orthonormal block Q_1, the block equivalent of Equations (1.5), (1.7) is

$$A[Q_1, \ldots, Q_m] = [Q_1, \ldots, Q_m]T_m + Q_{m+1}C_{m+1}^T[0, \ldots, 0, I] \tag{1.15}$$

where T_m is a symmetric block tridiagonal matrix:

$$T_m = \begin{pmatrix} B_1 & C_2 & & \\ C_2^T & B_2 & \ddots & \\ & \ddots & \ddots & C_m \\ & & C_m^T & B_m \end{pmatrix}. \tag{1.16}$$

The block structure introduces a few technical questions. Equation (1.15) directly constructs the product $R_{m+1} = Q_{m+1}C_{m+1}^T$, from which the orthonormal block Q_{m+1} is recovered; the choice of Q_{m+1} is not unique, however. If R_{m+1} is full rank, then any orthonormal basis can be chosen for Q_{m+1}; often a QR factorization is applied to R_{m+1}. If R_{m+1} is not full rank, then the block size may be reduced, or kept constant by padding Q_{m+1} with additional basis vectors. The block sizes may also be increased when necessary to resolve clustered eigenvalues.

Baglama et al. [3, 4] present a block Lanczos method with constant block sizes and implicit shifting at Leja points. Cullum [11] uses a hybrid procedure where the first block Q_1 has degree k, and all succeeding blocks Q_i are single vectors; in this variation, explicit orthogonalization against certain vectors in Q_1 is necessary. The ABLE method by Bai, Day, and Ye [5] combines a block nonsymmetric Lanczos method with a criterion for block enlargement and biorthonormality correction.

1.3 The finite element method

1.3.1 Overview

The *finite element method* is a discretization approach used in the numerical solution of differential equations. The general idea is to restrict the search space to one spanned by an easily-constructed set of basis functions. By using a variational principle, the solution to the differential equation is a linear combination of basis functions (or elements) that must satisfy a certain constraint when integrated against any single basis function. Taken together, this system of constraints form a matrix equation whose solutions approximate those of the original differential equation.

To describe this more concretely, suppose that for a given linear differential operator L, we are solving the nonhomogeneous equation $L(u) = f$. Suppose also that the desired solution u is contained within a known space $\mathcal{V}$. Then, we construct a finite-dimensional subspace $\mathcal{V}_h$ approximating $\mathcal{V}$; a space of continuous real-valued functions might be approximated by a space of piecewise-linear functions, for example. Generally, this finite-dimensional discretization is achieved by partitioning the domain Ω into an ordered mesh. Any function in $\mathcal{V}_h$ is then determined by its value on the edges and/or vertices of this mesh. Then, we can express an approximate solution $u_h = \sum_{i=1}^{N} x_i \phi_i$ as a linear combination of basis functions $\{\phi_1, \phi_2, \ldots, \phi_N\}$, which are chosen so that each basis function has support limited to a small number of mesh elements.

Since $L(u) = f$ everywhere inside the domain Ω, it follows that

$$\int_\Omega L(u)\phi \, dx = \int_\Omega f\phi \, dx \tag{1.17}$$

for any test function $\phi \in \mathcal{V}$. If $u_h \approx u$ is a good approximation, then we also have

$$\int_\Omega L(u_h)\phi_i \, dx \approx \int_\Omega f\phi_i \, dx \tag{1.18}$$

or equivalently

$$\sum_{j=1}^{N} \left(\int_\Omega L(\phi_j)\phi_i \, dx \right) x_j \approx \int_\Omega f\phi_i \, dx \tag{1.19}$$

for all $i = 1, 2, \ldots, N$. Each choice of i in Equation (1.19) gives rise to a linear equation in the coefficients $x_1, x_2, \ldots, x_N$. Therefore, this system is simply a matrix equation of order N of the form $Ax = b$, where $a_{ij} = \int_\Omega L(\phi_j)\phi_i \, dx$ and $b_i = \int_\Omega f\phi_i \, dx$.

A key practical feature of this approach is that A is generally a sparse matrix. With a natural choice of basis, the support of a basis element ϕ_i is restricted to a small number of adjacent mesh cells; therefore, a_{ij} is zero for all except a small set of elements $\{\phi_j\}$ whose support overlaps the support of ϕ_i. Additionally, for regular meshes and suitably symmetric bases, the nonzero elements of A are determined from a single stencil of overlapping elements. In this case, the stencil can be computed once and the entire matrix A completed inexpensively.

The chief difficulty is that the order N can be very large if a fine mesh is used, driving up the cost of solving the resulting system; several approaches are taken to reduce this cost. In most applications, the solution has more complicated behavior restricted to a relatively small portion of the domain. The order N is reduced by using an adaptively-sized mesh which is coarser in the areas of simpler behavior; of course, the classical stencil cannot be used with an adaptive mesh.

Further enhancements, *domain decomposition* and *multigrid*, are active areas of research. Domain decomposition partitions the domain in a number of pieces, each of which is modeled separately. The problem reduces to a series of subproblems that connect to the others through the subdomain boundaries; each subproblem is solved simultaneously on a parallel computer. Multigrid methods [39] approach the problem of determining mesh resolution by utilizing a series of coarse and fine meshes. After solving the problem to low accuracy with a coarse mesh, the mesh is refined where necessary and solved again in a "W"-shaped series of coarsenings and refinings. When the mesh refinement is determined based only on the discretized problem (rather than the underlying physical characteristics), then the method is called *algebraic multigrid.*

1.3.2 A finite element example

To illustrate the idea, consider solving the two-dimensional Poisson's equation

$$\begin{cases} -\Delta u = f & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega \end{cases} \tag{1.20}$$

where Ω is the square domain $(0,1) \times (0,1) \in \mathbb{R}^2$ with boundary $\partial\Omega$, and $f \in C^2(\Omega)$ is a given real-valued function. Let $\mathcal{V}$ be the space of C^2 functions on Ω which also attain zero on $\partial\Omega$; it is well known [18] that there exists a unique $u \in \mathcal{V}$ satisfying Equation (1.20).

For this problem, a natural choice would be to partition Ω into a uniform $n \times n$ mesh, with a node located at $(i/n, j/n)$ for each $i, j = 0, \ldots, n$. Each of the $(n-1)^2$ interior nodes are labeled $p_1, p_2, \ldots, p_N$. Additional edges are added to triangulate the mesh, as in Figure 1.3.

Next, we define a suitable finite-dimensional subspace $\mathcal{V}_h$ approximating the subspace $\mathcal{V}$. For the mesh chosen above, we choose $\mathcal{V}_h$ to be the space of continuous functions which attain zero on $\partial\Omega$ and are piecewise linear on each triangle. Such a function can be determined uniquely from its values on each of the interior nodes $\{p_1, \ldots, p_N\}$. Construct a set of piecewise linear functions $\{\phi_1, \ldots, \phi_N\}$ satisfying

$$\phi_i(p_j) = \begin{cases} 1, & i = j \\ 0, & i \neq j. \end{cases} \tag{1.21}$$

It is easy to see that this set $\{\phi_1, \ldots, \phi_N\}$ forms a natural basis of $\mathcal{V}_h$.

Lastly, we use the basis functions $\{\phi_i\}$ to convert the model equation (1.20) into

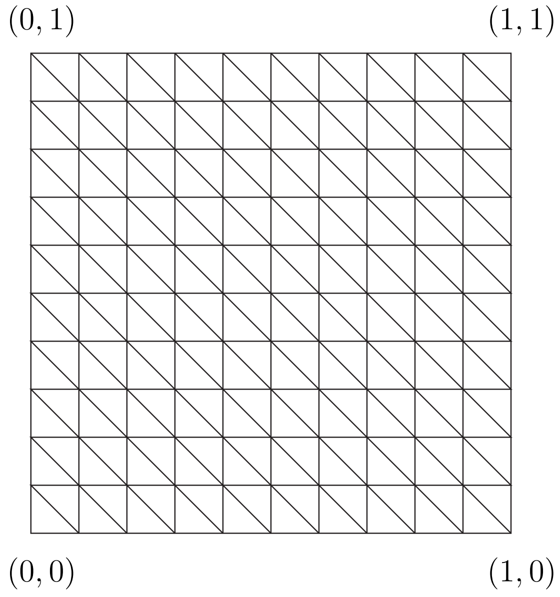


Figure 1.3: A triangulated square mesh, $n = 10$.

a matrix formulation. Since $f + \Delta u \equiv 0$ inside Ω, integrating this quantity against any test function will yield zero. In particular,

$$0 = \int_\Omega (f + \Delta u)\phi_i \, dx \tag{1.22}$$

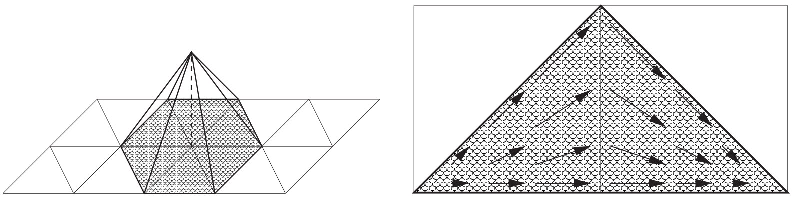
for any $i = 1, \ldots, N$.

Integrating by parts and using the boundary condition from (1.20) gives

$$\begin{aligned} 0 &= \int_\Omega (f + \Delta u)\phi_i \, dx \\ &= \int_\Omega f\phi_i \, dx - \int_\Omega \nabla u \cdot \nabla \phi_i \, dx + \int_{\partial\Omega} \phi_i \frac{\partial u}{\partial \nu} \, dS \\ &= \int_\Omega f\phi_i \, dx - \int_\Omega \nabla u \cdot \nabla \phi_i \, dx. \end{aligned} \tag{1.23}$$

At this point, we make the assumption that u is contained in the approximate subspace $\mathcal{V}_h$, and write $u \approx u_h = \sum_{j=1}^N x_j \phi_j$. Equation (1.23) thus gives the system of equations

$$\sum_{j=1}^N x_j \left(\int_\Omega \nabla \phi_j \cdot \nabla \phi_i \, dx \right) = \int_\Omega f\phi_i \, dx, \qquad i = 1, \ldots, N. \tag{1.24}$$



(a) The standard tent-shaped finite element

(b) A lowest-order Raviart-Thomas finite element

Figure 1.4: Two choices of finite element

This is the linear matrix equation $Ax = b$, where $a_{ij} = \int_\Omega \nabla\phi_j \cdot \nabla\phi_i \, dx$ and $b_i = \int_\Omega f\phi_i \, dx$.

1.3.3 Basis elements for the finite element method

The simplest type of space $\mathcal{V}_\langle$ to construct using finite elements would be the space of continuous functions which are piecewise-linear on each triangle of the mesh (as used in the previous example). The natural basis is then the set of piecewise-linear functions $\{\phi_i\}$ satisfying Equation (1.21). Each basis element ϕ_j corresponds to a node p_j of the given mesh, with support restricted to the adjacent triangles of the mesh. An illustration of this "tent"-shaped basis element appears in Figure 1.4(a).

An alternative type of finite element was introduced by Raviart and Thomas [46]. These elements are used to approximate the space of vector-valued functions

$$H(\text{div}, \Omega) = \left\{ \boldsymbol{u} \in [H(\Omega)]^n : \text{div } \boldsymbol{u} \in H(\Omega) \right\}. \tag{1.25}$$

Let $\mathcal{T}_h$ be the collection of triangles forming the mesh partition of Ω. The Raviart-Thomas space of order k is a function $\phi \in H(\text{div}, \Omega)$ satisfying two properties: on any triangle $T \in \mathcal{T}_h$, div ϕ is a polynomial of degree less than or equal to k, and the restriction of $\phi \cdot \nu_T$ to any side of T is a polynomial of degree less than or equal to k.

Therefore, the lowest-order Raviart-Thomas space is

$$\mathcal{V}_h = \left\{ \phi_h(\boldsymbol{x}) \in H(\mathrm{div}, \Omega) : \phi_h\big|_T \in \mathcal{P}_0^n + \mathcal{P}_o \boldsymbol{x},\ \forall T \in \mathcal{T}_h \right\} \tag{1.26}$$

i.e. the space of piecewise-constant functions and scalar multiples of $\boldsymbol{x}$. With this definition, it is possible to construct a basis of finite elements which correspond not to the nodes of the mesh, but rather to the edges; each finite element has nonzero flux across exactly one edge. The support of the finite element is restricted to the two triangles adjacent to its corresponding edge; the element has constant positive divergence on one triangle and constant negative divergence on the other. See Figure 1.4(b) for an illustration in the two-dimensional case.

In second-order elliptic applications, Raviart-Thomas finite elements are less likely than conventional piecewise-linear finite elements to produce spurious near-zero eigenmodes. A mixed method combining piecewise-linear and Raviart-Thomas elements in vibration analysis is discussed in Bermúdez [8]. Analysis indicating $O(h^2)$ convergence (for meshes with maximum edge length h) appears in Rodríguez [47] and Bermúdez and Durán [9]. Similar convergence results appear in Arbogast, Wheeler, and Yotov [2], where Raviart-Thomas elements are applied in a cell-centered finite difference model.

Chapter 2

Techniques for quadratic eigenproblems

2.1 Linearization

One approach to solving a matrix polynomial eigenvalue problem would be to convert it to an equivalent generalized or linear eigenvalue problem. Indeed, an arbitrary matrix polynomial $\lambda^k I + \lambda^{k-1} A_{k-1} + \cdots + \lambda A_1 + A_0$ has an eigenpair $(\hat{\lambda}, \hat{x})$ if and only if its companion matrix has a corresponding eigenpair:

$$\begin{pmatrix} & I & & \\ & & \ddots & \\ & & & I \\ -A_0 & -A_1 & \cdots & -A_{k-1} \end{pmatrix} \begin{pmatrix} \hat{x} \\ \hat{\lambda}\hat{x} \\ \vdots \\ \hat{\lambda}^{k-1}\hat{x} \end{pmatrix} = \hat{\lambda} \cdot \begin{pmatrix} \hat{x} \\ \hat{\lambda}\hat{x} \\ \vdots \\ \hat{\lambda}^{k-1}\hat{x} \end{pmatrix}. \tag{2.1}$$

This new eigenvalue problem could then be solved using an iterative method or direct solver.

Linearizations other than (2.1) are used in various applications. Equation (2.3) in the following section describes a linearization of a Hermitian quadratic eigenvalue problem as a Hermitian generalized eigenvalue problem. Mehrmann and Watkins [36] linearize a quadratic eigenvalue problem of the form $(\lambda^2 M + \lambda G + K)x = 0$, where M is symmetric positive definite, K is symmetric, and G is skew-symmetric, as the following skew-Hamiltonian/Hamiltonian matrix pencil:

$$\lambda \begin{pmatrix} M & G \\ 0 & M \end{pmatrix} \begin{pmatrix} \lambda x \\ x \end{pmatrix} = \begin{pmatrix} 0 & -K \\ M & 0 \end{pmatrix} \begin{pmatrix} \lambda x \\ x \end{pmatrix}.$$

Recall that a matrix A is *Hamiltonian* if $(AJ)^T = AJ$, where $J = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}$; similarly, A is *skew-Hamiltonian* if $(AJ)^T = -AJ$. An Arnoldi method called SHIRA is then applied to the skew-Hamiltonian/Hamiltonian pencil. Hwang et al. [29] apply the same linearization to $(\lambda^2 M + \lambda(G + \epsilon D) + K)x = 0$ where D is symmetric, by treating its linearized form as a perturbed skew-Hamiltonian/Hamiltonian system, applying SHIRA, and using the Jacobi-Davidson algorithm to correct for the perturbation. In another paper by Mehrmann and Watkins [37], an explicit linearization technique is given to linearize a matrix polynomial of arbitrary degree k. They show that if the coefficients of the original polynomial are alternately symmetric and skew-symmetric, then the system can be linearized into an order nk symmetric/skew-symmetric matrix pencil.

2.1.1 Tridiagonal-diagonal reduction

Recent work by Tisseur and others [54, 10] tackles the symmetric quadratic eigenvalue problem by reducing a linearization to a tridiagonal-diagonal form, then solving the reduced generalized eigenvalue problem. Suppose the quadratic problem is written as

$$(\lambda^2 M + \lambda C + K)x = 0. \tag{2.2}$$

Tisseur [10] suggests a linearization of Equation (2.2) which preserves symmetry, such as

$$A = \begin{pmatrix} 0 & K \\ K & C \end{pmatrix}, \qquad B = \begin{pmatrix} K & 0 \\ 0 & -M \end{pmatrix}. \tag{2.3}$$

Eigenvalues of Equation (2.2) are also eigenvalues of the generalized problem $Ax = \lambda Bx$. Note that A, B are symmetric but need not be definite. Indeed, if either A or B were definite, then it would follow that all eigenvalues were real (by taking a Cholesky factorization of the definite matrix), which we know not to be the case in general.

Therefore, we use an LDL^T factorization to write $P^TBP = LDL^T$, with D a diagonal matrix of 1×1 and 2×2 blocks. The eigendecomposition of D can be written as $D = X\Lambda X^T = X|\Lambda|^{1/2}J|\Lambda|^{1/2}X^T$, where J is a diagonal matrix with entries ±1 and X is block diagonal; the 1×1 blocks of X are equal to 1, and the 2×2 blocks are in the form of Jacobi rotations, $\begin{pmatrix} c & s \\ -s & c \end{pmatrix}$ with $c^2 + s^2 = 1$. Thus, with the choice $M = PL^{-t}X|\Lambda|^{-1/2}$, the symmetric-diagonal generalized pair (M^TAM, J) is congruent to (A, B) (assuming that B is nonsingular).

It remains to reduce $\hat{A} = M^TAM$ to a tridiagonal form, while keeping J diagonal. This problem can be viewed as tridiagonalizing the matrix $\hat{A}$ with respect to the J inner product, and the usual techniques (Givens rotations, Householder reflectors, and Lanczos method) generalize. A Givens-type approach (zeroing one entry of $\hat{A}$ at a time) would use Givens rotations whenever the corresponding entries of J have the same sign, and *hyperbolic transformations* of the form $\begin{pmatrix} c & -s \\ -s & c \end{pmatrix}$, $s^2 + c^2 = 0$ when the signs of J differ. Analogously, hyperbolic Householder reflectors can be defined by

$$H = P\left(J - \frac{2vv^T}{v^TJv}\right),$$

where P is a permutation and $v^TJv \neq 0$. Also, a variation of Lanczos method can be used, where all inner products are computed with respect to J.

When the original problem has been reduced to the tridiagonal-diagonal eigenvalue problem $(S - \lambda D)x = 0$, the eigenvalues of $T := D^{-1}S$ are computed by finding the roots of the characteristic polynomial $p(\lambda) = \det(T - \lambda I) = 0$. Starting with a vector $z^{(0)}$ of initial approximations to the roots of p, the Ehrlich-Aberth iteration

$$z_j^{(k+1)} = z_j^{(k)} - \frac{\frac{p(z_j^{(k)})}{p'(z_j^{(k)})}}{1 - \frac{p(z_j^{(k)})}{p'(z_j^{(k)})} \sum_{k \neq j} \frac{1}{z_j^{(k)} - z_k^{(k)}}}$$

converges at least linearly. In order to use this iteration effectively, one needs a reliable way to compute the Newton correction

$$\frac{p(\lambda)}{p'(\lambda)} = \frac{-1}{\text{trace}(T - \lambda I)^{-1}}.$$

Interestingly, this quantity can be computed stably and economically from the QR factorization of $T - \lambda I$ – see [10] for details.

2.2 Krylov subspace projections

The linearization approach has certain drawbacks, especially for large problems. The matrices involved have order kn, rather than n. Their eigenvectors are of the form $[x, \lambda x, \cdots, \lambda^{k-1}x]^T$, and therefore are poorly scaled when λ is very large or small; this may be undesirable numerically. Furthermore, a projection of a linearized problem need not preserve the properties of the original problem, such as symmetry, or possessing a field of values entirely in the left half-plane.

In this section, we focus on projection techniques that approximate a quadratic eigenvalue problem with another quadratic eigenvalue problem of lower dimension.

2.2.1 A Krylov subspace method for the monic QEP

In this section, we present a recent projection method [26, 34] to approximate a monic quadratic eigenvalue problem $(\lambda^2 I + \lambda B + C)x = 0$ with a lower-order quadratic eigenproblem

$$(\lambda^2 I_k + \lambda H_B + H_C)u = 0$$

where the $k \times k$ matrices H_B, H_C have a special form. An iterative method is described for constructing an orthonormal basis Q_k of a Krylov-type subspace so that $Q_k^T B Q_k = H_B, Q_k^T C Q_k = H_C$. Arnoldi-like and Lanczos-like versions of this method can be applied to nonsymmetric and symmetric quadratic matrix problems, respectively.

The Arnoldi-type process

Recall that applying $k-1$ steps of the Arnoldi method to a single matrix A and a starting vector q_1 produces an orthonormal basis $Q_k = \{q_1, q_2, \ldots, q_k\}$ of the Krylov space $\mathcal{K}_k = \text{span}\{q_1, Aq_1, \ldots, A^{k-1}q_k\}$ so that $H_k = Q_k^T A Q_k$ is an upper Hessenberg matrix (whose spectrum approximates that of A). In the quadratic case, we seek a similar orthonormal matrix Q_k that will reduce the matrices B, C to something Hessenberg-like. Actually reducing B and C to Hessenberg matrices H_B, H_C will be impossible in general; if Hessenberg H_B is desired, then the choice of initial vector q_1 determines Q_k up to sign, with no way to assure that $H_C = Q_k^T C Q_k$ has any particular structure.

However, it is possible to construct H_B, H_C with the following Hessenberg-like structure: the first columns of H_B, H_C contain two and three nonzeros respectively, and each successive column contains two fewer nonzeros than the previous column

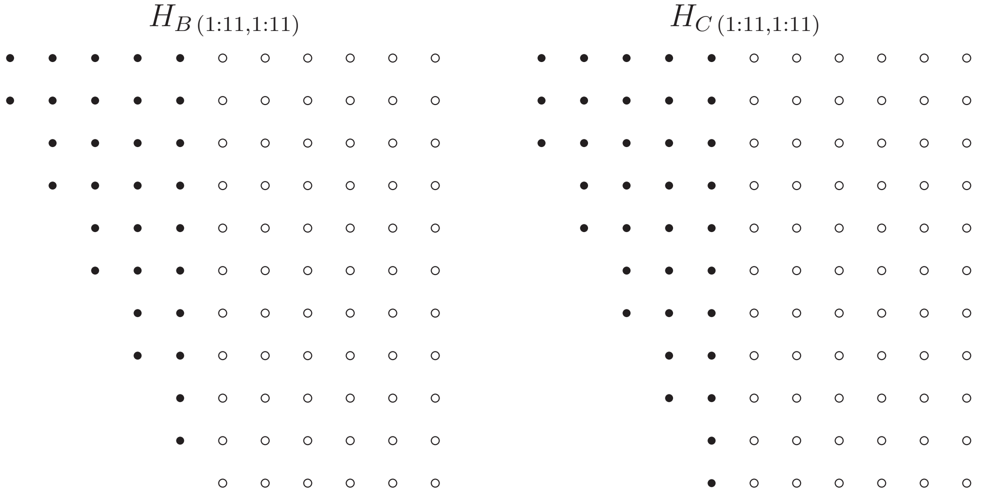


Figure 2.1: Partial structure of H_B, H_C from Arnoldi process

(Figure 2.1). Denoting the (i, j) entries of H_B and H_C by $h_{B;ij}$ and $h_{C;ij}$ respectively, we have the following:

Lemma 2.2.1. *There exists an orthogonal $n \times n$ matrix Q such that $Qe_1 = e_1$, $h_{B;ij} = 0$ for all $i \geq 2j + 1$, and $h_{C;ij} = 0$ for all $i \geq 2j + 2$.*

Proof. The proof is constructive. In each of the following matrix decompositions, let $*$ denote an unknown element or submatrix. Write

$$B = \begin{pmatrix} b_{11} & * \\ b_1 & * \end{pmatrix}.$$

Choose an orthogonal $(n-1) \times (n-1)$ matrix $\hat{Q}_{1b}$ such that $\hat{Q}_{1b}^T b_1 = \beta_1 e_1$ (a Householder reflector is one such matrix). Writing $Q_{1b} = \begin{pmatrix} 1 & 0 \\ 0 & \hat{Q}_{1b} \end{pmatrix}$, we have

$$Q_{1b}^T B Q_{1b} = \left(\begin{array}{c|c} b_{11} & * \\ \hline \beta_1 & * \\ 0 & * \end{array}\right), \qquad Q_{1b}^T C Q_{1b} = \left(\begin{array}{c|c} c_{11} & * \\ \hline c_{21} & * \\ c_1 & * \end{array}\right).$$

Similarly, choose an orthogonal $(n-2) \times (n-2)$ matrix $\hat{Q}_{1c}$ so that $\hat{Q}_{1c}^T c_1 = \gamma_1 e_1$.

Writing $Q_{1c} = \begin{pmatrix} I_2 & 0 \\ 0 & \hat{Q}_{1c} \end{pmatrix}$ and letting $Q_1 = Q_{1b}Q_{1c}$ gives

$$Q_1^T B Q_1 = \left(\begin{array}{c|c} b_{11} & * \\ \beta_1 & * \\ \hline 0 & * \\ 0 & * \end{array}\right), \qquad Q_1^T C Q_1 = \left(\begin{array}{c|c} c_{11} & * \\ c_{21} & * \\ \hline \gamma_1 & * \\ 0 & * \end{array}\right).$$

This completes the reduction of the first columns of B and C. Continuing, partition $Q_1^T B Q_1$ as

$$Q_1^T B Q_1 = \left(\begin{array}{cc|c} * & * & * \\ * & * & * \\ 0 & b_{32} & * \\ \hline 0 & b_2 & * \end{array}\right)$$

and find an orthogonal $(n-3) \times (n-3)$ matrix $\hat{Q}_{2b}$ so that $\hat{Q}_{2b}^T b_2 = \beta_2 e_1$. Writing $Q_{2b} = \begin{pmatrix} I_3 & 0 \\ 0 & \hat{Q}_{2b} \end{pmatrix}$ gives

$$Q_{2b}^T Q_1^T B Q_1 Q_{2b} = \left(\begin{array}{c|c|c} * & * & * \\ * & * & * \\ 0 & b_{32} & * \\ \hline 0 & \beta_2 & * \\ 0 & 0 & * \end{array}\right), \quad Q_{2b} Q_1^T C Q_1 Q_{2b} = \left(\begin{array}{c|c|c} * & * & * \\ * & * & * \\ * & * & * \\ \hline 0 & c_{42} & * \\ 0 & c_2 & * \end{array}\right).$$

Choose an orthogonal $(n-4) \times (n-4)$ matrix $\hat{Q}_{2c}$ so that $\hat{Q}_{2c}^T c_2 = \gamma_2 e_1$. Letting $Q_{2c} = \begin{pmatrix} I_4 & 0 \\ 0 & \hat{Q}_{2c} \end{pmatrix}$ and $Q_2 = Q_{2b}Q_{2c}$ as before, we have

$$Q_2^T Q_1^T B Q_1 Q_2 = \left(\begin{array}{c|c|c} * & * & * \\ * & * & * \\ 0 & b_{32} & * \\ 0 & \beta_2 & * \\ \hline 0 & 0 & * \\ 0 & 0 & * \end{array}\right), \quad Q_2 Q_1^T C Q_1 Q_2 = \left(\begin{array}{c|c|c} * & * & * \\ * & * & * \\ * & * & * \\ 0 & c_{42} & * \\ \hline 0 & \gamma_2 & * \\ 0 & 0 & * \end{array}\right).$$

This completes the reduction of the first two columns. By continuing similarly for k steps, we can find an orthogonal matrix $Q = Q_k Q_{k-1} \cdots Q_2 Q_1$ so that the claim holds for the first k columns of $H_B = Q^T B Q^T$ and $H_C = Q^T C Q_T$. It is easy to see that $Qe_1 = e_1$. Repeat for $k = \lfloor \frac{n-1}{2} \rfloor$ iterations to complete the decomposition. $\square$

As a corollary, given any unit-length vector q_1 we can extend it to an orthogonal matrix $Q = [q_1, q_2, \ldots, q_n]$ so that $H_B = Q^T BQ$ and $H_C = Q^T CQ$ have the desired form. Let V be an orthogonal matrix with q_1 as its first column. By Lemma 2.2.1, there exists an orthogonal $\tilde{Q}$ so that $H_B = \tilde{Q}^T(V^T BV)\tilde{Q}$ and $H_C = \tilde{Q}^T(V^T CV)\tilde{Q}$; then $Q = V\tilde{Q}$ satisfies $Qe_1 = q_1$.

Therefore, since an appropriate Q does exist, we can use the identities

$$BQ = QH_B,\ CQ = QH_C \tag{2.4}$$

to construct a recurrence generating Q, H_B, H_C from a given starting vector q_1. Examining the j-th column of Equations (2.4) gives the recurrences

$$Bq_j = \sum_{i=1}^{2j-1} q_i h_{B;ij} + q_{2j} h_{B;2j,j}, \tag{2.5}$$

$$Cq_j = \sum_{i=1}^{2j} q_i h_{C;ij} + q_{2j+1} h_{C;2j+1,j}. \tag{2.6}$$

Thus, starting with $Q_{2j-1} = [q_1, q_2, \ldots, q_{2j-1}]$, we can compute q_{2j}, q_{2j+1}, and the corresponding columns of H_B, H_C by applying a Gram-Schmidt process successively to Equation (2.5) and then to Equation (2.6). The complete algorithm appears in Figure 2.2, with a modification as discussed in the next section.

The low-rank case

Observe that according to the construction in Lemma 2.2.1, the lower bandwidth of H_B, H_C increases rapidly. However, the construction neglects the possibility of a “nice” breakdown, which can be used to keep the bandwidth from growing too quickly. Suppose by way of example that c_1 turns out to be zero after applying the

```
Input: $\|q_1\|_2 = 1$, $k > 0$
$N = 1$
for $j = 1, 2, \ldots, k$ do
    if $j > N$ then
        exit
    end if
    {Compute column $j$ of $H_B$}
    $\hat{q} = Bq_j$
    for $i = 1, 2, \ldots, N$ do
        $h_{B;ij} = q_i^T \hat{q}$
        $\hat{q} = \hat{q} - q_i h_{B;ij}$
    end for
    $h_{B;N+1,j} = \|\hat{q}\|_2$
    if $h_{B;N+1,j} > 0$ then
        $N = N + 1$
        $q_N = \hat{q}/h_{B;Nj}$
    end if
    {Compute column $j$ of $H_C$}
    $\hat{q} = Cq_j$
    for $i = 1, 2, \ldots, N$ do
        $h_{C;ij} = q_i^T \hat{q}$
        $\hat{q} = \hat{q} - q_i h_{C;ij}$
    end for
    $h_{C;N+1,j} = \|\hat{q}\|_2$
    if $h_{C;N+1,j} > 0$ then
        $N = N + 1$
        $q_N = \hat{q}/h_{C;Nj}$
    end if
end for
```

Figure 2.2: Arnoldi-type algorithm

first orthogonal transformation Q_{1b}. That is, we have

$$Q_{1b}^T B Q_{1b} = \left(\begin{array}{c|c} b_{11} & * \\ \hline \beta_1 & * \\ 0 & * \end{array}\right), \qquad Q_{1b}^T C Q_{1b} = \left(\begin{array}{c|c} c_{11} & * \\ \hline c_{21} & * \\ 0 & * \end{array}\right).$$

This is actually a good situation, since we may now reduce the bandwidth. Instead of applying a suitable reflection Q_{1c} to zero out as much of the first column of C as possible, we can proceed directly to the second columns. The reflection Q_{2b} we choose to act on the second column of B need only fix the first two entries in each column instead of the first three:

$$Q_{2b}^T Q_{1b}^T B Q_{1b} Q_{2b} = \left(\begin{array}{c|c|c} * & * & * \\ * & * & * \\ \hline 0 & \beta_2 & * \\ 0 & 0 & * \end{array}\right), \quad Q_{2b} Q_{1b}^T C Q_{1b} Q_{2b} = \left(\begin{array}{c|c|c} * & * & * \\ * & * & * \\ \hline 0 & c_{32} & * \\ 0 & c_2 & * \end{array}\right).$$

Similarly, the reflection Q_{2c} which reduces the second column of C need only fix the first three entries in each column. Writing $Q_2 = Q_{2b} Q_{2c}$ we have

$$Q_2^T Q_{1b}^T B Q_{1b} Q_2 = \left(\begin{array}{c|c|c} * & * & * \\ * & * & * \\ 0 & \beta_2 & * \\ \hline 0 & 0 & * \\ 0 & 0 & * \end{array}\right), \quad Q_2 Q_{1b}^T C Q_{1b} Q_2 = \left(\begin{array}{c|c|c} * & * & * \\ * & * & * \\ 0 & c_{32} & * \\ \hline 0 & \gamma_2 & * \\ 0 & 0 & * \end{array}\right).$$

Thus, the reduction continues to termination as before, with each subsequent column of H_B, H_C containing one fewer nonzero element after the breakdown.

The corresponding modification to the iterative algorithm is as follows. At any point in the algorithm, let N be the number of q-vectors that have been computed so far; $N = 1$ initially. If no breakdown occurs in the application of Gram-Schmidt to the product Bq_j, then the following variation of Equation (2.5) determines the next basis vector q_{N+1}:

$$Bq_j - \sum_{i=1}^{N} q_i h_{B;ij} = q_{N+1} h_{B;N+1,j}.$$

Therefore, the basis $Q = \{q_1, \ldots, q_{N+1}\}$ is enlarged, and its current size N incremented. If a breakdown occurs, then no new q-vector needs to be computed, and N is left alone. Likewise, after computing $\hat{r} = Cq_j$, we orthogonalize $\hat{r}$ against the current basis with

$$Cq_j - \sum_{i=1}^{N} q_i h_{C;ij} = q_{N+1} h_{C;N+1,j}$$

and compute the next q-vector unless breakdown occurs. The complete algorithm with this modification appears in Figure 2.2.

Reduction in the lower bandwidth of H_B, H_C is significant, since the eigenvalues of the projected QEP $(\mu^2 I_k + \mu H_B + H_C)u = 0$ converge slowly when the bandwidth is large. As the following theorem shows, the lower bandwidth does stay small if a low-rank linear combination of B and C exists.

Theorem 2.2.2. *If a nonzero linear combination $E = xB + yC$ has rank d, then the lower bandwidth of each projected matrix H_B, H_C is not more than $d + 1$.*

Proof. Consider the case when $y \neq 0$; the case when B has rank d is handled similarly. Intuitively, at step j the lower bandwidth can increase by at most 1, and only if there is no breakdown in the orthogonalization of Bq_j and Cq_j. After orthogonalizing Bq_j, we have $\mathrm{span}\{q_1, \ldots, q_N\} = \mathrm{span}\{q_1, \ldots, q_{N-1}, Bq_j\}$. Now $Cq_j \notin \mathrm{span}\{q_1, \ldots, q_N\}$ if and only if $Eq_j \notin \mathrm{span}\{q_1, \ldots, q_N\}$; it follows that breakdown can fail to occur at most d times.

To show this more formally, for each $j = 1, 2, \ldots$ let α_j, β_j be the number of nonzeros in column j of H_B, H_C respectively. We have $\alpha_1 \leq \beta_1 \leq \alpha_2 \leq \ldots$, with successive terms differing by at most 1. At step k we have $BQ_{(:,1:k)} = Q_{(:,1:\alpha_k)} H_{B(1:\alpha_k,1:k)}$

and $CQ_{(:,1:k)} = Q_{(:,1:\beta_k)} H_{C(1:\beta_k,1:k)}$, so we can write

$$EQ_{(:,1:k)} = Q_{(:,1:\beta_k)}(xH_{B(1:\beta_k,1:k)} + yH_{C(1:\beta_k,1:k)}).$$

Therefore, the rank of $W = xH_{B(1:\beta_k,1:k)} + yH_{C(1:\beta_k,1:k)}$ is at most d. Next, we bound the rank below by the number of times the reduction of C does not break down. Let $i_1 < i_2 < \cdots < i_l$ be the set of indices where breakdown does not occur: $\alpha_{i_j} < \beta_{i_j}$ for all $j = 1, 2, \ldots, l$. It follows that $h_{B;\beta_{i_j},i_j} = 0$, and so the (β_{i_j}, i_j) entries of W are nonzero for $j = 1, 2, \ldots, l$. Also, the sequence $\{\beta_{i_j}\}$ is strictly increasing. So, the $i_1, i_2, \ldots, i_l$-th columns of W are linearly independent; the rank of W must be at least l.

Thus, we have $l \leq d$. We know that $\alpha_{j+1} \leq \alpha_j + 2$, and equality implies $j \in \{i_1, i_2, \ldots, i_l\}$. So

$$\begin{aligned}\alpha_j &= \alpha_1 + \sum_{i=1}^{j-1}(\alpha_{i+1} - \alpha_i)\\ &\leq \alpha_1 + (j-1) + l\\ &\leq j + l + 1 \leq j + d + 1.\end{aligned}$$

Similarly, $\beta_j \leq j + d + 1$. □

A Lanczos-type process

Observe that if the Arnoldi-type recurrence is applied to a symmetric QEP, then the projections $T_B = Q_k^T B Q_k$, $T_C = Q_k^T C Q_k$ of B, C onto the subspace Q_k are also symmetric. Therefore, the zero subdiagonal elements are accompanied by corresponding zero superdiagonals, as in Figure 2.3. This is described as a Lanczos-type process, and the projections labeled T_B, T_C by way of analogy.

The corresponding modifications to the algorithm in Figure 2.2 are minimal, al-

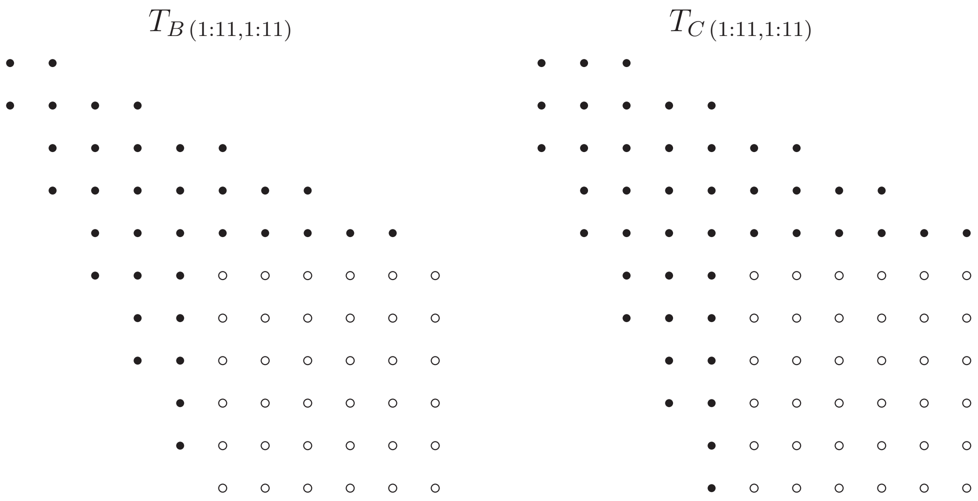


Figure 2.3: Structure of T_B, T_C from Lanczos process

though some additional housekeeping is necessary to count the extra zero entries. The basic recurrence is now given by

$$Bq_j = \sum_{i=l_b}^{2j-1} q_i t_{B;ij} + q_{2j} t_{B;2j,j}, \tag{2.7}$$

$$Cq_j = \sum_{i=l_c}^{2j} q_i t_{C;ij} + q_{2j+1} t_{C;2j+1,j}, \tag{2.8}$$

where l_b, l_c are the smallest indices such that $\alpha_{l_b}, \beta_{l_c} \geq j$. Therefore, in each iteration of the algorithm we save α_j, β_j, and increment l_b, l_c when necessary. The complete Lanczos-type algorithm appears in Figure 2.4.

When the matrices B, C are not symmetric, the Lanczos-type process can be extended to an analogue of the nonsymmetric Lanczos method [26, pp. 10–11]. As in nonsymmetric Lanczos, this is achieved by sacrificing the orthogonality of the basis upon which each matrix is projected. Instead, we may construct biorthogonal bases V, W so that $T_B = V^T BW$ and $T_C = V^T CW$ are nonsymmetric matrices with the nonzero structure appearing in Figure 2.3. To show this, observe that the

```
Input: $\|q_1\|_2 = 1$, $k > 0$
$N = 1$; $\alpha_1 = 1$; $\beta_1 = 1$
$l_b = 1$; $l_c = 1$
for $j = 1, 2, \ldots, k$ do
    if $j > N$ then
        exit
    end if
    {Compute column $j$ of $T_B$}
    $\hat{q} = Bq_j$
    if $j > \alpha_{l_b}$ then
        $l_b = l_b + 1$
    end if
    for $i = l_b, \ldots, N$ do
        $t_{B;ij} = q_i^T \hat{q}$
        $\hat{q} = \hat{q} - q_i t_{B;ij}$
    end for
    $t_{B;N+1,j} = \|\hat{q}\|_2$
    if $t_{B;N+1,j} > 0$ then
        $N = N + 1$; $\alpha_j = N$
        $q_N = \hat{q}/t_{B;Nj}$
    end if
    {Compute column $j$ of $T_C$}
    $\hat{q} = Cq_j$
    if $j > \beta_{l_c}$ then
        $l_c = l_c + 1$
    end if
    for $i = l_c, \ldots, N$ do
        $t_{C;ij} = q_i^T \hat{q}$
        $\hat{q} = \hat{q} - q_i t_{C;ij}$
    end for
    $t_{C;N+1,j} = \|\hat{q}\|_2$
    if $t_{C;N+1,j} > 0$ then
        $N = N + 1$; $\beta_j = N$
        $q_N = \hat{q}/t_{C;Nj}$
    end if
end for
```

Figure 2.4: Symmetric Lanczos-type algorithm

constructive proof in Lemma 2.2.1 hinges on the fact that for any vector x, there exists an orthogonal matrix Q (i.e. a Householder reflector) so that Qx is a multiple of e_1. By applying a similarity transformation containing Q to B, C, the undesired entries of each column in turn are zeroed out (the Arnoldi-type process). When B, C are symmetric, the same similarity transformation also zeros out part of the corresponding row, resulting in the Lanczos-type reduction in Figure 2.3.

For a nonsymmetric method, we need to find a similarity transformation (not necessarily orthogonal) which will zero out the undesired entries of a column and its corresponding row simultaneously. This is possible because of the following lemma (also stated explicitly in [27]):

Lemma 2.2.3. *Let x, y be real n-vectors with $x^T y \neq 0$. There exists a nonsingular matrix W so that $W^{-1}x$ is a multiple of e_1 and $y^T W$ is a multiple of e_1^T.*

Proof. First, construct an orthogonal Q so that $Qx = \alpha e_1$ and $Qy = \beta e_1 + \gamma e_2$, for appropriate constants α, β, γ. Such a Q is easy to construct with a product of Householder reflectors

$$Q = \left(I - \frac{(x - e_1)(x - e_1)^T}{\|x - e_1\|^2}\right)\left(I - \frac{(z - e_2)(z - e_2)^T}{\|z - e_2\|^2}\right) \tag{2.9}$$

for $z = y - \frac{x^T y}{x^T x}x$ (assuming $z \neq e_2$ and $x \neq e_1$; otherwise, simplify appropriately). It follows that for the desired choice of W, $(W^{-1}Q^T)e_1$ is a multiple of e_1 and $(Qy)^T$ is a multiple of $e_1^T(W^{-1}Q^T)$. This is satisfied for the choice of W such that

$$W^{-1}Q^T = \left(\begin{array}{cc|c} 1 & \gamma/\beta & \\ & 1 & \\ \hline & & I \end{array}\right). \tag{2.10}$$

The condition $x^T y \neq 0$ is sufficient (though not strictly necessary) to ensure that $\beta \neq 0$. □

Using Lemma 2.2.3, it is possible to reproduce the construction in Lemma 2.2.1, substituting appropriate nonorthogonal similarity transformations instead of orthogonal. Suppose we have performed k steps of the nonsymmetric reduction, yielding W so that

$$W^{-1}BW = \left(\begin{array}{c|c} (T_B)_k & \begin{array}{c} 0 \\ y_1^T \end{array} \\ \hline \begin{array}{cc} 0 & x_1 \end{array} & * \end{array}\right), \quad W^{-1}CW = \left(\begin{array}{c|c} (T_C)_k & \begin{array}{c} 0 \\ * \end{array} \\ \hline \begin{array}{cc} 0 & * \end{array} & * \end{array}\right).$$

Assuming $x_1^T y_1 \neq 0$, find $\hat{W}_1$ according to Lemma 2.2.3 so that $\hat{W}_1^T y_1$ and $\hat{W}_1^{-1} x_1$ are multiples of e_1. Writing $W_1 = W \begin{pmatrix} I & \\ & \hat{W}_1 \end{pmatrix}$,

$$W_1^{-1}BW_1 = \left(\begin{array}{cc|cc} \multicolumn{2}{c|}{\multirow{2}{*}{$(T_B)_k$}} & 0 & 0 \\ & & b_1 & 0 \\ \hline 0 & b_2 & * & * \\ 0 & 0 & * & * \end{array}\right), \quad W_1^{-1}CW_1 = \left(\begin{array}{cc|cc} \multicolumn{2}{c|}{\multirow{2}{*}{$(T_C)_k$}} & 0 & 0 \\ & & c_1 & y_2^T \\ \hline 0 & c_2 & * & * \\ 0 & x_2 & * & * \end{array}\right).$$

Similarly, construct $\hat{W}_2$ so that $\hat{W}_2^T y_2$ and $\hat{W}_2^{-1} x_2$ are multiples of e_1; setting $W_2 = W_1 \begin{pmatrix} I & \\ & \hat{W}_2 \end{pmatrix}$ gives

$$W_2^{-1}BW_2 = \left(\begin{array}{cc|cc|c} \multicolumn{2}{c|}{\multirow{2}{*}{$(T_B)_k$}} & 0 & 0 & 0 \\ & & b_1 & 0 & 0 \\ \hline 0 & b_2 & * & * & * \\ 0 & 0 & * & * & * \\ \hline 0 & 0 & * & * & * \end{array}\right), \quad W_2^{-1}CW_2 = \left(\begin{array}{cc|cc|c} \multicolumn{2}{c|}{\multirow{2}{*}{$(T_C)_k$}} & 0 & 0 & 0 \\ & & c_1 & c_3 & 0 \\ \hline 0 & c_2 & * & * & * \\ 0 & c_4 & * & * & * \\ \hline 0 & 0 & * & * & * \end{array}\right).$$

Thus, as long as the pairs $\{x_i, y_i\}$ are not orthogonal (corresponding to a "bad" breakdown in nonsymmetric Lanczos), biorthogonal bases V, W can be constructed. The defining recurrences are then

$$Bw_j = \sum_{i=l_b}^{2j-1} w_i t_{B;ij} + w_{2j} t_{B;2j,j}, \qquad B^T v_j = \sum_{i=l_b}^{2j-1} v_i t_{B;ji} + v_{2j} t_{B;j,2j}, \tag{2.11}$$

$$Cw_j = \sum_{i=l_c}^{2j} w_i t_{C;ij} + w_{2j+1} t_{C;2j+1,j}, \qquad C^T v_j = \sum_{i=l_c}^{2j} v_i t_{C;ji} + v_{2j+1} t_{C;j,2j+1}. \tag{2.12}$$

2.2.2 Extension to nonmonic polynomials

The next question is how to generalize these Krylov-type methods to quadratic eigenvalue problems of the form $(\lambda^2 A + \lambda B + C)x = 0$, where A is not necessarily I. We assume that A is nonsingular, however. The original QEP can then be replaced by the equivalent problem

$$(\lambda^2 I + \lambda A^{-1}B + A^{-1}C)x = 0 \tag{2.13}$$

which clearly has the same eigenpairs as the original. This is the straightforward transformation used in 2.5. Of course, the matrices $A^{-1}B, A^{-1}C$ need not be computed explicitly (and should not for numerical stability). Instead, any matrix-vector product $y = (A^{-1}B)x$ is replaced by a two-step process: compute $z = Bx$, then solve $Ay = z$ for y. This introduces a one-time cost of factoring A into a suitable form for use in the linear solves.

If the original QEP is symmetric and A is positive definite, then it is similarly possible to construct a monic symmetric QEP with the same spectrum. Take the Cholesky factorization of $A = LL^T$, and solve the equivalent QEP

$$(\lambda^2 I + \lambda\tilde{B} + \tilde{C})\tilde{x} = 0, \tag{2.14}$$

where $\tilde{B} = L^{-1}BL^{-T}$ and $\tilde{C} = L^{-1}CL^{-T}$. Once the initial Cholesky factorization is computed, a matrix-vector product $\tilde{B}v, \tilde{C}v$ can be computed (relatively) inexpensively by performing two triangular solves and one ordinary matrix-vector product. In the case when A is large and sparse, the appropriate choice of permutation to maintain sparsity of the Cholesky factorization was discussed in Section 1.2.1.

We apply the Lanczos-type process from Figure 2.4 to a symmetric QEP; the argument works identically for the Arnoldi-type process. Applying Algorithm 2.4 to Equation 2.14 and running to termination produces an orthogonal $n \times n$ matrix Q_n so that

$$Q_n^T(\lambda^2 I + \lambda\tilde{B} + \tilde{C})Q_n = (\lambda^2 I + \lambda\tilde{T}_B + \tilde{T}_C) \tag{2.15}$$

or equivalently

$$(Q_n^T L^{-1})(\lambda^2 A + \lambda B + C)(L^{-T}Q_n) = (\lambda^2 I + \lambda\tilde{T}_B + \tilde{T}_C). \tag{2.16}$$

Thus $W_n = L^{-T}Q_n$ is an A-symmetric basis such that $W_n^T(\lambda^2 A + \lambda B + C)W_n$ is reduced to the desired banded form. We restrict our attention to the monic case, applying the above transformations to the applications in Chapter 4 when necessary.

2.3 Subspace projection for the monic QEP

Consider the monic quadratic eigenvalue problem

$$(\lambda^2 I + \lambda B + C)x = 0. \tag{2.17}$$

For any subspace $S \subseteq \mathbb{C}^n$, a Ritz pair consists of a scalar Ritz value μ and a nonzero Ritz vector $u \in S$, such that the residual of (2.17) satisfies the following Petrov-Galerkin condition

$$(\mu^2 I + \mu B + C)u \perp S. \tag{2.18}$$

If the columns of Q_k form an orthonormal basis of S, then we have an equivalent formulation of the problem; μ is a Ritz value with corresponding Ritz vector $Q_k u$ if and only if

$$(\mu^2 I_k + \mu B_k + C_k)u = 0 \tag{2.19}$$

where B_k, C_k are the projections of B, C onto S given by $B_k = Q_k^* B Q_k$ and $C_k = Q_k^* C Q_k$. Note that if an exact eigenvector x of the full-dimensional problem (2.17) is contained in S, then the corresponding Ritz pair is exact; this is readily seen since $x = Q_k Q_k^T x$ and

$$\begin{aligned}(\lambda^2 I_k + \lambda B_k + C_k) Q_k^T x &= Q_k^T (\lambda^2 I + \lambda B + C) Q_k Q_k^T x \\ &= Q_k^T (\lambda^2 I + \lambda B + C) x \\ &= 0.\end{aligned}$$

Therefore, we expect that when the angle between x and S is small, the Ritz pairs will provide good approximations to the exact eigenpairs. To show this, we use the following result from Saad [49, pp. 130–131]:

Theorem 2.3.1. *Let M be an $n \times n$ (possibly nonsymmetric) matrix with the eigenpair (λ, u). Also let V_k be an $n \times k$ orthonormal matrix. Then*

$$\|(V_k^T M V_k - \lambda I) V_k^T u\|_2 \leq \gamma \|(I - V_k V_k^T) u\|_2 \tag{2.20}$$

where $\gamma = \|V_k^T M (I - V_k V_k^T)\|_2$. Equivalently, one can write the conclusion as

$$\frac{\|(V_k^T M V_k - \lambda I) V_k^T u\|_2}{\|V_k^T u\|} \leq \gamma \tan \theta(u, S) \tag{2.21}$$

where S is the subspace spanned by the columns of V_k.

Apply Theorem 2.3.1 to a linearization of the desired quadratic eigenvalue problem. Suppose (λ, x) is an eigenpair of interest of the QEP $(\lambda^2 I + \lambda B + C)x = 0$, and let Q_k be an appropriate orthonormal basis of the subspace S. Then construct suitable M, V_k, u:

$$M = \begin{pmatrix} 0 & I \\ -C & -B \end{pmatrix}, \qquad V_k = \begin{pmatrix} Q_k & \\ & Q_k \end{pmatrix}, \qquad u = \begin{pmatrix} x \\ \lambda x \end{pmatrix}. \tag{2.22}$$

Write S_V to denote the subspace spanned by the columns of V_k. It follows that

$$\begin{aligned}
\|V_k u\| &= \sqrt{1+|\lambda|^2}\,\|Q_k^T x\|, \\
\tan\theta(u, S_V) &= \frac{(I - V_k V_k^T)u}{\|V_k^T u\|} \\
&= \frac{\sqrt{1+|\lambda|^2}\,\|(I - Q_k Q_k^T)x\|}{\sqrt{1+|\lambda|^2}\,\|Q_k^T x\|} \\
&= \tan\theta(x, S), \\
\left\|(V_k^T M V_k - \lambda I)V_k^T u\right\| &= \left\| \left(\begin{pmatrix} 0 & I_k \\ -C_k & -B_k \end{pmatrix} - \lambda I \right) \begin{pmatrix} Q_k^T x \\ \lambda Q_k^T x \end{pmatrix} \right\| \\
&= \left\|(\lambda^2 I_k + \lambda B_k + C_k) Q_k^T x\right\|,
\end{aligned}$$

and

$$\gamma = \|Q_k^T C(I - Q_k Q_k^T), Q_k^T B(I - Q_k Q_k^T)\|. \tag{2.23}$$

Therefore,

$$\frac{\|(\lambda^2 I_k + \lambda B_k + C_k) Q_k^T x\|}{\|Q_k^T x\|} \le \sqrt{1+\|\lambda\|^2}\,\gamma \tan\theta(x, S). \tag{2.24}$$

Thus, the residual of the normalized eigenpair $(\lambda, Q_k x/\|Q_k x\|)$, interpreted as an approximate solution to the projected QEP, is small when x is close to S.

Unfortunately, if the problem is ill-conditioned, a small residual is not sufficient to conclude that the Ritz value is accurate. Roughly speaking, for a simple eigenvalue, the residual error is magnified by the condition number of the eigenvalue [49, p. 93]. Again, we utilize the linearization M of the QEP, as in (2.22). For a given eigenvalue $\lambda(M)$, the condition number of λ is defined as

$$\mathrm{Cond}(\lambda) = \frac{1}{\cos\langle x_M, y_M \rangle} \tag{2.25}$$

where x_M, y_M are the left and right eigenvectors corresponding to λ. These eigenvectors relate to those of the QEP as follows. As observed before, the right eigenvector of

M is given by $x_M = \begin{pmatrix} x \\ \lambda x \end{pmatrix}$, where x is a corresponding right eigenvector of the QEP. It is easy to check that a left eigenvector of M corresponding to λ is $y_M = \begin{pmatrix} -C^* y \\ \bar{\lambda} y \end{pmatrix}$. The condition number of $\lambda(M)$ is therefore

$$\begin{aligned}
\text{Cond}(\lambda) &= \frac{\|x_M\|\|y_M\|}{|y_M^* x_M|} && (2.26) \\
&= \frac{(\sqrt{1+|\lambda|^2}\|x\|)(\sqrt{\|C^*y\|^2 + |\lambda|^2\|y\|^2})}{|\lambda^2 y^* x - y^* C x|} && (2.27) \\
&= \frac{\sqrt{1+|\lambda|^2}\sqrt{(\|C^*y\|/\|y\|)^2 + |\lambda|^2}}{|\lambda^2 - (y^*Cx/y^*x)|} \cdot \frac{\|x\|\|y\|}{|y^*x|}. && (2.28)
\end{aligned}$$

Equation (2.28) gives a relationship between the condition number of λ and the angle between the corresponding left and right eigenvectors x, y of the QEP. This angle may not be available in general; in the special case when the QEP is symmetric, the eigenvectors satisfy $y = \bar{x}$ and

$$\frac{\|x\|\|y\|}{|x^T y|} = \frac{\|x\|^2}{|x^T x|}. \tag{2.29}$$

Thus, in the symmetric case, an eigenvalue λ of the QEP will be ill-conditioned if its corresponding eigenvector x is close to *quasi-null* (i.e. $x \neq 0$ but $x^T x = 0$).

2.3.1 Arnoldi variant I

One approach is to project the quadratic problem onto a subspace generated by the Arnoldi method. If we are interested in estimating a large eigenvalue λ, then we can write the eigenproblem as

$$\left(\lambda I + B + \frac{1}{\lambda} C\right) x = 0 \tag{2.30}$$

where the lowest-order term $\frac{1}{\lambda}C$ is expected to be small. Therefore, a subspace Q_k containing a good approximate eigenvector x of the linear problem $(\lambda I + B)x = 0$

```
Input: $\|q_0\|_2 = 1$, $k > 0$
for $j = 1, \ldots, k$ do
    $r = Bq_{j-1}$
    {Compute $q_j$ by Gram-Schmidt}
    for $m = 1, \ldots, j-1$ do
        $r = r - (q_m^* r)q_m$
    end for
    $q_j = r/\|r\|$
end for
{Find large Ritz values $\mu$}
$B_k = Q_k^* B Q_k$ {Hessenberg}
$C_k = Q_k^* C Q_k$
Solve $(\mu^2 I_k + \mu B_k + C_k)u_k = 0$.
```

Figure 2.5: Arnoldi variant I

is a likely place to look for an eigenvector of the quadratic problem. Applying k steps of the Arnoldi method to the matrix B produces such a subspace Q_k. After Q_k has been constructed, the projected problem in Equation (2.19) is solved to get Ritz values approximating the large eigenvalues of Equation (2.17).

A restarted formulation of this method incorporating a shift and invert can produce fast convergence. One way to reduce the cost of inverting the quadratic eigenvalue problem is to perform the necessary solves inexactly. Good results have been obtained in [58] by using a preconditioned GMRES method to perform the solves inexactly, even if the selected tolerance is large.

2.3.2 Arnoldi variant II

Similarly, an Arnoldi method can be used to generate a subspace containing an approximate eigenvector corresponding to a small eigenvalue λ. Looking again at the original eigenvalue problem as stated in Equation (2.17), the middle term λB is expected to be small when λ is small. By neglecting this middle term, the eigenvalue

```
Input: $\|q_0\|_2 = 1$, $k > 0$
for $j = 1, \ldots, k$ do
    $r = Cq_{j-1}$
    {Compute $q_j$ by Gram-Schmidt}
    for $m = 1, \ldots, j-1$ do
        $r = r - (q_m^* r)q_m$
    end for
    $q_j = r/\|r\|$
end for
{Find small Ritz values $\mu$}
$B_k = Q_k^* B Q_k$
$C_k = Q_k^* C Q_k$ {Hessenberg}
Solve $(\mu^2 I_k + \mu B_k + C_k)u_k = 0$.
```

Figure 2.6: Arnoldi variant II

problem is approximated by the linear problem $(\lambda^2 I + C)x = 0$; applying k steps of the Arnoldi method to C produces a subspace Q_k, as in the Arnoldi variant described in the previous section.

A detailed description of Arnoldi variants I and II appear in Figure 2.5 and Figure 2.6.

2.3.3 Arnoldi variant III

In Arnoldi variants I and II, we produced a linear eigenvalue problem approximating the quadratic eigenvalue problem $(\lambda^2 I + \lambda B + C)x = 0$ by neglecting either the λB term or the constant term C from the matrix polynomial. If λ is expected to be large, then C is negligible (Arnoldi variant I); if λ is small, then λB is negligible relative to C (Arnoldi variant II). However, when λ is small, then $\lambda^2 I$ is in fact the least significant term by analogy with Arnoldi variant I. Therefore, neglecting the leading term gives a generalized eigenvalue problem

$$(\lambda B + C)x = 0,$$

```
**Input:** $\|q_0\|_2 = 1$, $k > 0$
for $j = 1, \ldots, k$ do
    $r = B^{-1} C q_{j-1}$
    $\{$Compute $q_j$ by Gram-Schmidt$\}$
    for $m = 1, \ldots, j-1$ do
        $r = r - (q_m^* r) q_m$
    end for
    $q_j = r / \|r\|$
end for
$\{$Find small Ritz values $\mu\}$
$B_k = Q_k^* B Q_k$
$C_k = Q_k^* C Q_k = B_k H_k$
Solve $(\mu^2 I_k + \mu B_k + C_k) u_k = 0$.
```

Figure 2.7: Arnoldi variant III

from which a subspace Q_k is generated as before by applying k steps of the Arnoldi method to the matrix $B^{-1}C$. Note that neither of the projected matrices B_k, C_k has Hessenberg structure; however, it is readily seen that $C_k = B_k H_k$, where H_k is the Hessenberg matrix generated by Arnoldi. See Figure 2.7.

2.4 Structure-preserving shift and invert

Next, we consider accelerating the Lanczos-type algorithm from Figure 2.4 by incorporating a complex shift-and-invert transformation. Given a real symmetric QEP $(\lambda^2 A + \lambda B + C)x = 0$ and an approximate complex eigenvalue σ, we seek another real symmetric QEP $(\mu^2 \hat{A} + \mu \hat{B} + \hat{C})u = 0$ whose eigenvalues correspond to the eigenvalues of the original problem through a spectral transformation mapping $\lambda_1 \approx \sigma$ to a large eigenvalue μ_1. For a reasonable choice of σ, the transformed problem will then have a dominant and well-separated eigenvalue μ_1, which is expected to converge quickly under our Krylov-type method. Thus, there are two questions: how to construct a suitable shifted problem economically, and how to adapt the algorithm in Figure 2.4

for a general quadratic matrix equation.

2.4.1 A simple shift approach

A natural choice would be the simple shift $\mu = \frac{1}{\lambda - \sigma}$. Shifting the QEP $(\lambda^2 A + \lambda B + C)x = 0$ by σ directly and inverting gives the equivalent QEP

$$(\mu^2(\sigma^2 A + \sigma B + C) + \mu(2\sigma A + B) + A)x = 0.$$

Unfortunately, this QEP is complex symmetric for a complex shift σ, and therefore the transfomation is not structure preserving in general. Instead, we will perform the shift in two steps; first shift by the real part of σ as above, then separate the real and imaginary parts of the matrix equation in order to perform the imaginary shift. Letting $\sigma = a + bi$, shifting by a gives

$$(s^2 A_1 + sB_1 + C_1)x = 0$$

where $A_1 = A$, $B_1 = 2aA + B$, $C_1 = a^2 A + aB + C$, and $s = \lambda - a$. Next, shift by the imaginary part bi to get

$$\begin{aligned} s^2 A_1 + sB_1 + C_1 &= [(s-bi)^2 + 2sbi + b^2]A_1 + [(s-bi) + bi]B_1 + C_1 \\ &= [t^2 A_1 + tB_1 + (C_1 - b^2 A_1)] + i[2btA_1 + bB_1] \end{aligned} \tag{2.31}$$

where $t = s - bi = \lambda - \sigma$.

The following lemma allows us to replace this complex matrix polynomial with a real symmetric polynomial of twice the order.

Lemma 2.4.1. *Suppose $\{M_1, N_2, \ldots, M_k, N_k\}$ are real $n \times n$ matrices. Define a real $2n \times 2n$ matrix polynomial $R(\lambda)$ and a complex $n \times n$ matrix polynomial $C(\lambda)$ by*

$$R(\lambda) = \sum_{j=0}^{k} \lambda^j \begin{pmatrix} M_j & N_j \\ N_j & -M_j \end{pmatrix}$$

$$C(\lambda) = \sum_{j=0}^{k} \lambda^j [M_j + iN_j].$$

Then each eigenvalue λ of C corresponds to a conjugate pair of eigenvalues $\lambda, \bar{\lambda}$ of R.

Proof. Observe that the $2n \times 2n$ matrix $\begin{pmatrix} I & iI \\ iI & I \end{pmatrix}$ is nonsingular. Therefore, the matrix polynomial

$$\begin{aligned} M(\lambda) &= \begin{pmatrix} I & iI \\ iI & I \end{pmatrix} R(\lambda) \begin{pmatrix} I & iI \\ iI & I \end{pmatrix} \\ &= \sum_{j=0}^{k} \lambda^j \begin{pmatrix} I & iI \\ iI & I \end{pmatrix} \begin{pmatrix} M_j & N_j \\ N_j & -M_j \end{pmatrix} \begin{pmatrix} I & iI \\ iI & I \end{pmatrix} \\ &= \sum_{j=0}^{k} \lambda^j \begin{pmatrix} 2(M_j + iN_j) & \\ & -2(M_j - iN_j) \end{pmatrix} \end{aligned}$$

has the same eigenvalues as R.

If λ is an eigenvalue of C with corresponding eigenvector u, it follows immediately that $\lambda, \begin{pmatrix} u \\ 0 \end{pmatrix}$ is an eigenpair of M. Therefore, λ is an eigenvalue of R; $\bar{\lambda}$ is also an eigenvalue of R since $R(\lambda)x = 0$ iff $R(\bar{\lambda})\bar{x} = 0$.

Conversely, if λ is an eigenvalue of R, then λ is an eigenvalue of M. Let $\begin{pmatrix} u \\ v \end{pmatrix}$ be a corresponding eigenvector of M; we have

$$\begin{aligned} 0 &= M(\lambda) \begin{pmatrix} u \\ v \end{pmatrix} \\ &= \sum_{j=0}^{k} \lambda^j \begin{pmatrix} 2(M_j + iN_j) & \\ & -2(M_j - iN_j) \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} \\ &= \begin{pmatrix} 2\sum_{j=0}^{k} \lambda^j (M_j + iN_j) u \\ -2\sum_{j=0}^{k} \lambda^j (M_j - iN_j) v \end{pmatrix}. \end{aligned}$$

Therefore $\sum_{j=0}^{k} \lambda^j (M_j + iN_j) u = 0$, and by taking conjugates, $\sum_{j=0}^{k} \bar{\lambda}^j (M_j + iN_j) \bar{v} = 0$. Since u, v are not both zero, at least one of $\lambda, \bar{\lambda}$ is an eigenvalue of C. □

With this lemma, we can replace the complex polynomial in (2.31) by the real symmetric polynomial

$$t^2\begin{pmatrix} A_1 & \\ & -A_1 \end{pmatrix} + t\begin{pmatrix} B_1 & 2bA_1 \\ 2bA_1 & -B_1 \end{pmatrix} + \begin{pmatrix} C_1 & bB_1 \\ bB_1 & -C_1 \end{pmatrix}.$$

Dividing by t^2 and substituting gives the desired quadratic shifted polynomial

$$\mu^2\begin{pmatrix} C_1 & bB_1 \\ bB_1 & -C_1 \end{pmatrix} + \mu\begin{pmatrix} B_1 & 2bA_1 \\ 2bA_1 & -B_1 \end{pmatrix} + \begin{pmatrix} A_1 & \\ & A_1 \end{pmatrix}$$

or in terms of the original matrices A, B, C:

$$\begin{aligned} &\mu^2\begin{pmatrix} a^2A + aB + C & 2abA + bB \\ 2abA + bB & -(a^2A + aB + C) \end{pmatrix} \\ &+ \mu\begin{pmatrix} 2aA + B & 2bA \\ 2bA & -(2aA + B) \end{pmatrix} + \begin{pmatrix} A & \\ & -A \end{pmatrix}. \end{aligned} \tag{2.32}$$

All eigenvalues μ of the polynomial in (2.32) occur in conjugate pairs. Furthermore, each eigenvalue λ of the original system is obtained from shifting one of the eigenvalues in the pair $\mu, \bar{\mu}$; that is, either $\lambda = \sigma + (1/\mu)$ or $\lambda = \sigma + (1/\bar{\mu})$.

2.4.2 Saad-Parlett double shift

While the simple shift in the preceding section produces a real symmetric matrix polynomial, it doubles the dimension of the QEP undesirably; therefore, we consider other possibilities. The goal is to transform the original real symmetric QEP into another real symmetric QEP, where the desired eigenvalues are shifted to the exterior of the spectrum and are well-separated (for better performance in Krylov methods).

Saad and Parlett [44] discuss approaches to solving the generalized nonsymmetric eigenvalue problem $Fu = \lambda Mu$ (with F, M real) by inverse iteration, while minimizing the use of complex arithmetic. For a complex shift σ, inverse iteration about σ is equivalent to applying the power method directly to $(F - \sigma M)^{-1}M$. The key observation is that applying the conjugate of this matrix is the same as shifting about $\bar{\sigma}$; therefore, we can construct purely real operators

$$\hat{B}_+ = \frac{1}{2}\big((F-\sigma M)^{-1}M + (F-\bar{\sigma}M)^{-1}M\big) = \mathrm{Re}[(F-\bar{\sigma}M)^{-1}M] \tag{2.33}$$

$$\hat{B}_- = \frac{1}{2i}\big((F-\sigma M)^{-1}M - (F-\bar{\sigma}M)^{-1}M\big) = \mathrm{Im}[(F-\bar{\sigma}M)^{-1}M]. \tag{2.34}$$

If (λ, u) is an eigenpair of the generalized eigenvalue problem, then it is easy to check that u is an eigenvector of both $\hat{B}_+$ and $\hat{B}_-$ with respective eigenvalues μ_+, μ_- satisfying

$$\mu_+ = \frac{1}{2}\left(\frac{1}{\lambda-\sigma} + \frac{1}{\lambda-\bar{\sigma}}\right) \tag{2.35}$$

$$\mu_- = \frac{1}{2i}\left(\frac{1}{\lambda-\sigma} - \frac{1}{\lambda-\bar{\sigma}}\right). \tag{2.36}$$

To apply this spectral transformation to the quadratic eigenvalue problem $(\lambda^2 I + \lambda B + C)x = 0$, use the equivalent linearization $F = \begin{pmatrix} 0 & I \\ -C & -B \end{pmatrix}$ and $M = I_{2n}$. The operator of interest is then

$$\begin{aligned}(F-\sigma M)^{-1}M &= \left\{\begin{pmatrix} 0 & I \\ -C & -B \end{pmatrix} - \sigma I\right\}^{-1} \\ &= \begin{pmatrix} \frac{1}{\sigma}(L(\sigma)^{-1}C - I) & -L(\sigma)^{-1} \\ L(\sigma)^{-1}C & -\sigma L(\sigma)^{-1} \end{pmatrix} \end{aligned} \tag{2.37}$$

where $L(\sigma) = \sigma^2 I + \sigma B + C$. Writing $L(\sigma)^{-1} = L_R + iL_I$ and $\sigma = a + bi$, the real and imaginary parts of Equation (2.37) are the operators

$$\hat{B}_+ = \begin{pmatrix} \frac{a}{|\sigma|^2}(L_R C - I) + \frac{b}{|\sigma|^2}L_I C & -L_R \\ L_R C & bL_I - aL_R \end{pmatrix} \tag{2.38}$$

$$\hat{B}_- = \begin{pmatrix} \frac{b}{|\sigma|^2}(I - L_R C) + \frac{a}{|\sigma|^2}L_I C & -L_I \\ L_I C & -bL_R - aL_I \end{pmatrix}. \tag{2.39}$$

Recall that a 2×2 block matrix can be converted to an equivalent quadratic eigenvalue problem under mild assumptions. If $\left(\lambda, \begin{pmatrix} x \\ y \end{pmatrix}\right)$ is an eigenpair of $\begin{pmatrix} P & Q \\ R & S \end{pmatrix}$ where

$x \neq 0$ and Q is nonsingular, then (λ, x) is an eigenpair of the QEP

$$[\lambda^2 Q^{-1} - \lambda(SQ^{-1} + Q^{-1}P) + (SQ^{-1}P - R)]x = 0.$$

Therefore, the operators $\hat{B}_+$, $\hat{B}_-$ give rise to corresponding transformed quadratic eigenvalue problems satisfying (2.35)–(2.36):

$$\Big\{\mu_+^2(L_R^{-1}) + \mu_+ \left[\frac{a}{|\sigma|^2}(L_R^{-1} - C) - \frac{b}{|\sigma|^2} L_R^{-1} L_I C + aI - bL_I L_R^{-1}\right] + \frac{1}{|\sigma|^2}\left[b^2 L_R C + a^2 I - abL_I L_R^{-1} + b^2 L_I L_R^{-1} L_I C\right]\Big\}\, x = 0 \quad (2.40)$$

$$\Big\{\mu_-^2(L_I^{-1}) + \mu_- \left[\frac{b}{|\sigma|^2}(L_I^{-1} L_R C - L_I^{-1}) - \frac{a}{|\sigma|^2} C + aI + bL_R L_I^{-1}\right] + \frac{1}{|\sigma|^2}\left[b^2 L_I C - abI + b^2 L_R L_I^{-1} L_R C + b^2 L_R L_I^{-1}\right]\Big\}\, x = 0. \quad (2.41)$$

Assuming the desired λ is close to σ and far from $\bar{\sigma}$ (or vice versa), both of these spectral transformations possess a large, well-separated eigenvalue μ. If σ is close to the real axis, then $\mu_+ \approx 1/(\lambda - \sigma)$ is a better choice. Applying the operators L_R, L_I to a real vector v can be done by solving $L(\sigma)z = v$ in complex arithmetic and taking the real and imaginary parts of z, but this is expensive. Also, symmetry in the original eigenvalue problem may not be preserved.

2.4.3 Symmetry-preserving double shift

An alternative construction utilizes a double shift similar to (2.36):

$$\mu = \frac{1}{(\lambda - \sigma)(\lambda - \bar{\sigma})} = \frac{1}{(\lambda - a)^2 + b^2}. \quad (2.42)$$

We will construct a suitable shifted QEP by applying a series of symmetry-preserving transformations to the original QEP. Since it is easy to construct a symmetric transformation of a symmetric QEP that performs a real spectral shift ($\lambda \to \lambda - c$)

or a spectral inversion ($\lambda \to \frac{1}{\lambda}$), we express the transformation (2.42) as the composition of the following steps:

- Translate $\lambda \to \lambda - a$.
- Transform $\lambda \to \lambda^2$.
- Translate again $\lambda \to \lambda + b^2$.
- Invert $\lambda \to \frac{1}{\lambda}$.

The composition of these transformations yields the desired eigenvalue problem whose eigenvalues μ correspond to the original eigenvalues λ through the spectral transformation (2.42). The first, third, and fourth steps are straightforward real shifts and inversions; next, we show how the QEP is transformed to perform the spectral mapping $\lambda \to \lambda^2$.

Transforming $\boldsymbol{\lambda \to \lambda^2}$

Suppose (λ, x) is an eigenpair of the quadratic eigenvalue problem $(\lambda^2 A + \lambda B + C)x = 0$. For a nonsingular matrix M, consider the equation

$$(\lambda^2 A - \lambda B + C)M(\lambda^2 A + \lambda B + C)x = 0,$$

whose eigenvalues agree up to sign with those of the original QEP. Multiplying, this is equivalent to

$$\begin{aligned}[][\lambda^4 AMA + \lambda^2(AMC + CMA - BMB) + CMC \\ +\lambda^3(AMB - BMA) + \lambda(CMB - BMC)]x = 0.\end{aligned} \tag{2.43}$$

Therefore, the polynomial equation

$$[s^2 AMA + s(AMC + CMA - BMB) + CMC]y = 0 \tag{2.44}$$

has an eigenpair $s = \lambda^2$, $y = x$ if M is chosen so that

$$[\lambda^2(AMB - BMA) + (CMB - BMC)]x = 0. \tag{2.45}$$

Since $BM(\lambda^2 A + \lambda B + C)x = 0$, this condition simplifies to $(\lambda^2 A + \lambda B + C)MBx = 0$; if λ is a nondegenerate eigenvalue of the original QEP, then Equation (2.45) holds iff x is an eigenvector of MB. Clearly $M = B^{-1}$ is a choice which will hold for any eigenvector x; since a quadratic eigenvalue problem could have up to $2n$ distinct eigenvectors, this may be the only suitable constant M (up to scale). This is illustrated by the following example.

Consider the quadratic eigenvalue problem

$$\left\{\lambda^2 \begin{pmatrix} 1 & \\ & 1 \end{pmatrix} + \lambda \begin{pmatrix} 2 & 1 \\ 1 & -2 \end{pmatrix} + \begin{pmatrix} 2 & -2 \\ -2 & 0 \end{pmatrix}\right\} x = 0.$$

It is easy to check that the eigenvalues are $\lambda = -2, 2, i, -i$, with corresponding eigenvectors $\begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ i \end{pmatrix}, \begin{pmatrix} 1 \\ -i \end{pmatrix}$. If M is chosen so that $\begin{pmatrix} 2 \\ 1 \end{pmatrix}$ and $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$ are eigenvectors of MB, then

$$MB \begin{pmatrix} 2 & \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 2 & \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \alpha & \\ & \beta \end{pmatrix}. \tag{2.46}$$

However,

$$MB \begin{pmatrix} 1 \\ i \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta i + \frac{1}{2}(\alpha - \beta) \end{pmatrix} \tag{2.47}$$

which is a multiple of $\begin{pmatrix} 1 \\ i \end{pmatrix}$ only if $\alpha = \beta$. Then $M = \alpha B^{-1}$.

Thus using $M = B^{-1}$, a real symmetric QEP can be constructed according to Equation (2.42) through a series of four spectral transformations.

Complete transformation

For completeness, here we present an explicit construction of real symmetric $n \times n$ matrices $\hat{A}, \hat{B}, \hat{C}$ so that the eigenvalues of

$$(\mu^2 \hat{A} + \mu \hat{B} + \hat{C})u = 0$$

correspond to those of $(\lambda^2 A + \lambda B + C)x = 0$ through the spectral transformation in Equation (2.42). The QEP $(\lambda^2 A + \lambda B + C)x = 0$ is equivalent to the shifted problem $(\lambda_1^2 A_1 + \lambda_1 B_1 + C_1)x = 0$ where

$$\lambda_1 = \lambda - a \tag{2.48}$$

$$A_1 = A \tag{2.49}$$

$$B_1 = 2aA + B \tag{2.50}$$

$$C_1 = a^2 A + aB + C. \tag{2.51}$$

This is further equivalent to $(\lambda_2^2 A_2 + \lambda_2 B_2 + C_2)x = 0$ with

$$\begin{aligned}
\lambda_2 &= \lambda_1^2 = (\lambda - a)^2 \\
A_2 &= AB_1^{-1}A \\
B_2 &= -2a^2 AB_1^{-1}A + AB_1^{-1}C + CB_1^{-1}A - B \\
C_2 &= a^2 B + 2aC + (C - a^2 A)B_1^{-1}(C - a^2 A).
\end{aligned}$$

Lastly, shift again by b^2 to obtain the QEP $[(\lambda_2 + b^2)^2 A_3 + (\lambda_2 + b^2) B_3 + C_3]x = 0$, which is equivalent to the desired eigenvalue problem $(\mu^2 \hat{A} + \mu \hat{B} + \hat{C})x = 0$ with

$$\begin{aligned} \hat{A} = C_3 = (a^2 + b^2)^2 A B_1^{-1} A - (a^2 + b^2)[A B_1^{-1} C + C B_1^{-1} A - B] \\ + 2aC + C B_1^{-1} C \end{aligned} \tag{2.52}$$

$$\hat{B} = B_3 = -2(a^2 + b^2) A B_1^{-1} A + A B_1^{-1} C + C B_1^{-1} A - B \tag{2.53}$$

$$\hat{C} = A_3 = A B_1^{-1} A. \tag{2.54}$$

Note that if the original matrices A, B, C are symmetric, then the matrices $\hat{A}, \hat{B}, \hat{C}$ defined by Equations (2.50), (2.52)–(2.54) are also symmetric.

2.5 A residual-maximizing approach

The Rayleigh-Ritz methods described in the previous section work as follows: construct a subspace, project the eigenvalue problem onto that subspace, and then extract a Ritz value from the projected problem that approximates the desired eigenvalue. In the process, vectors $q_1, q_2, \dots, q_k$ spanning the k-dimensional subspace are constructed one by one. Thus, a sequence of nested subspaces

$$\begin{aligned} Q_1 &= \operatorname{span}\{q_1\}, \\ Q_2 &= \operatorname{span}\{q_1, q_2\}, \\ &\vdots \\ Q_k &= \operatorname{span}\{q_1, q_2, \dots, q_k\} \end{aligned}$$

is constructed before any Ritz pairs are computed. A natural question arises: can projections of the QEP onto the intermediate subspaces $Q_1, Q_2, \dots, Q_{k-1}$ be used to help select the subsequent basis vectors?

2.5.1 General description

Let $L(\lambda)$ denote the matrix polynomial $\lambda^2 A + \lambda B + C$. The eigenvalue problem is that of finding a scalar λ so that $L(\lambda)$ is singular. Using a simple shift σ and letting $\mu = \frac{1}{\lambda - \sigma}$, we have the shifted matrix polynomial $\hat{L}(\mu) = \mu^2 \hat{A} + \mu \hat{B} + \hat{C}$ with

$$\begin{aligned} \hat{A} &= \sigma^2 A + \sigma B + C = L(\sigma) \\ \hat{B} &= 2\sigma A + B \\ \hat{C} &= A. \end{aligned}$$

It follows that $\frac{1}{\mu^2}\hat{L}(\mu) = L(\lambda)$. The shifted eigenvalue problem can then be converted into the monic problem

$$(\mu^2 I + \mu \hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C})u = 0. \tag{2.55}$$

For each subspace of dimension j with $Q_j = [q_1, \ldots, q_j]$, we may project Equation (2.55) onto Q_j and compute the largest Ritz value μ_j (best approximating the eigenvalue closest to the shift σ) and the corresponding Ritz vector u_j. Specifically, (μ_j, u_j) satisfy

$$(\mu_j^2 I_j + \mu_j B_j + C_j)u_j = 0 \tag{2.56}$$

where $B_j = Q_j^*(\hat{A}^{-1}\hat{B})Q_j$ and $C_j = Q_j^*(\hat{A}^{-1}\hat{C})Q_j$. This series of Ritz pairs $\{(\mu_j, u_j)\}$ of projected eigenproblems may be used during the iteration.

2.5.2 A natural utilization of the order *j* Ritz pair

With the additional information provided by the Ritz pair (μ_j, u_j), how best can we choose the next basis vector $q_{j+1} \perp Q_j$? Our approach is the following. Any choice of q_{j+1} would give rise to a new projected eigenvalue problem

$$(\mu_{j+1}^2 I_{j+1} + \mu_{j+1} B_{j+1} + C_{j+1}) u_{j+1} = 0. \tag{2.57}$$

The Ritz pair (μ_j, u_j) satisfying Equation (2.56) can be interpreted naturally as an approximate solution $(\tilde{\mu}_{j+1}, \tilde{u}_{j+1})$ to Equation (2.57), defined by

$$\tilde{\mu}_{j+1} = \mu_j, \quad \tilde{u}_{j+1} = \begin{pmatrix} u_j \\ 0 \end{pmatrix}, \tag{2.58}$$

with a residual

$$r_{j+1} = (\tilde{\mu}_{j+1}^2 I_{j+1} + \tilde{\mu}_{j+1} B_{j+1} + C_{j+1}) \tilde{u}_{j+1} \tag{2.59}$$

(note that we have not yet actually constructed B_{j+1}, C_{j+1}).

Next, we set out the following principle determining the construction of q_{j+1} once the choice of approximate solution $(\tilde{\mu}_{j+1}, \tilde{u}_{j+1})$ has been made. Observe that if the residual r_{j+1} is small, then $(\tilde{\mu}_{j+1}, \tilde{u}_{j+1})$ is close to an exact Ritz pair of Equation (2.57). Therefore, we would not expect the largest Ritz value of Equation (2.57) to differ very much from that of Equation (2.56); enlarging the subspace by q_{j+1} offers little improvement. Therefore, we choose q_{j+1} orthogonal to Q_j so as to make $\|r_{j+1}\|$ as large as possible. In particular, with the choice of approximate solution from Equation (2.58), the residual is given by

$$\begin{aligned} r_{j+1} &= Q_{j+1}^*[\mu_j^2 I + \mu_j \hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C}] Q_{j+1} \begin{pmatrix} u_j \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} Q_j^* \\ q_{j+1}^* \end{pmatrix} [\mu_j^2 I + \mu_j \hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C}] Q_j u_j \\ &= \begin{pmatrix} 0 \\ q_{j+1}^*(\mu_j \hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C}) Q_j u_j \end{pmatrix} \end{aligned}$$

so that up to scale, q_{j+1} equals the restriction of $(\mu_j \hat{A}^{-1}\hat{B} + A^{-1}\hat{C}) Q_j u_j$ to the orthogonal complement of Q_j. It is easy to see that the restriction is nonzero unless

```
**Input:** $\|q_1\|_2 = 1$, $\sigma \in \mathbb{C}$, $k > 0$
Shift and invert $(\lambda^2 A + \lambda B + C)x = 0$ to get equivalent problem $(\mu^2 \hat{A} + \mu \hat{B} + \hat{C})x = 0$, $\mu = \frac{1}{\lambda - \sigma}$.
Let $\hat{L}(\mu) = \mu^2 I + \mu \hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C}$.
**for** $j = 1, 2, \ldots, k-1$ **do**
    Solve $Q_j^* \hat{L}(\mu_j) Q_j u_j = 0$. {Compute largest Ritz pair}
    Choose approximate solution $(\tilde{\mu}_{j+1}, \tilde{u}_{j+1})$; typically set $\tilde{\mu}_{j+1} = \mu_j$, $\tilde{u}_{j+1} = \begin{pmatrix} u_j \\ 0 \end{pmatrix}$. See Section 2.5.3.
    Let $r = Q_{j+1}^* \hat{L}(\tilde{\mu}_{j+1}) Q_{j+1} \tilde{u}_{j+1}$.
    Choose $q_{j+1} \perp Q_j$ maximizing $r$.
    Normalize $q_{j+1}$.
**end for**
Solve $Q_k^* \hat{L}(\mu_k) Q_k u_k = 0$.
```

Figure 2.8: Residual maximization algorithm

(μ_j, u_j) is an exact eigenpair of Equation (2.55), i.e. convergence has occurred.

The heart of this approach lies in the construction of the approximate solution to Equation (2.57). The approximate solution does not have to be chosen according to Equation (2.58); there are a number of reasonable constructions, as we will see in the next section. Each construction produces a new algorithm, with distinct convergence behavior. In this sense, one might characterize this algorithm as a residual-maximizing framework, rather than a single method. A complete description of the method appears in Figure 2.8.

2.5.3 Selection of approximate solution

In the previous section, one approximate solution

$$(\tilde{\mu}_{j+1}, \tilde{u}_{j+1}) = \left(\mu_j, \begin{pmatrix} u_j \\ 0 \end{pmatrix}\right)$$

was chosen. In this section, we consider some other possible choices of approximate solution, and show that some of these approximate solution choices are equivalent to algorithms discussed elsewhere.

Theorem 2.5.1. *Let $(\lambda^2 A+\lambda B+C)x = 0$ be a quadratic eigenvalue problem, with the equivalent shifted and inverted problem $(\mu^2 I+\mu\hat{A}^{-1}\hat{B}+\hat{A}^{-1}\hat{C})x = 0$. Applying each of the following algorithms to the shifted problem is equivalent to applying Algorithm 2.8 to the original problem with the specified choice of approximate solution.*

1. *Arnoldi variant I (Figure 2.5): $(\tilde{\mu}_{j+1}, \tilde{u}_{j+1}) = (\infty, e_j)$.*

2. *Arnoldi variant II (Figure 2.6): $(\tilde{\mu}_{j+1}, \tilde{u}_{j+1}) = (0, e_j)$.*

3. *Arnoldi-type Krylov process (Figure 2.2): $(\tilde{\mu}_{j+1}, \tilde{u}_{j+1}) = \begin{cases} (0, e_{j/2}), & j \text{ even} \\ (\infty, e_{\lceil j/2 \rceil}), & j \text{ odd.} \end{cases}$*

Proof. For each part, we need to show that both methods will produce the same subspace span$\{q_1, q_2, \ldots, q_k\}$ when the same starting vector q_1 is used. It is sufficient to show that the bases Q_k and $\hat{Q}_k$ produced by each method differ only by scale, i.e. $\hat{Q}_k = Q_k D$ for some nonsingular diagonal matrix D. We prove this by induction; if $Q_j, \hat{Q}_j$ are bases generated by each method which differ only by scale, we only need to show that continuing another step produces vectors q_{j+1}, $\hat{q}_{j+1}$ which are scalar multiples of each other.

1. Suppose Q_j is the basis produced by the first j iterations of Algorithm (2.8), with $(\tilde{\mu}_{j+1}, \tilde{u}_{j+1}) = (\infty, e_j)$. As Algorithm 2.8 implicitly assumes that $\tilde{\mu}_{j+1}$ is finite, we take $\tilde{\mu}_{j+1} = \infty$ to mean that q_{j+1} is defined as the limit of $q_{j+1}(\tilde{\mu})$ produced by the algorithm as $\tilde{\mu} \to \infty$ (assuming this limit exists and is nonzero).

For large $\tilde{\mu}$, the residual is

$$\begin{aligned} r &= Q_{j+1}^*(\tilde{\mu}^2 I + \tilde{\mu}\hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C})Q_{j+1}e_j \\ &= \begin{pmatrix} Q_j^*(\tilde{\mu}^2 I + \tilde{\mu}\hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C})q_j \\ q_{j+1}^*(\tilde{\mu}\hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C})q_j \end{pmatrix} \\ &= \begin{pmatrix} Q_j^*(\tilde{\mu}^2 I + \tilde{\mu}\hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C})q_j \\ \tilde{\mu}q_{j+1}^*(\hat{A}^{-1}\hat{B} + \frac{1}{\tilde{\mu}}\hat{A}^{-1}\hat{C})q_j \end{pmatrix}. \end{aligned}$$

Therefore, $q_{j+1}(\tilde{\mu})$ equals the projection onto $Q_j^{\perp}$ (up to scale)

$$(I - Q_jQ_j^*)(\hat{A}^{-1}\hat{B} + \frac{1}{\tilde{\mu}}\hat{A}^{-1}\hat{C})q_j. \tag{2.60}$$

Letting $\tilde{\mu} \to \infty$, we see that q_{j+1} is a scalar multiple of $(I - Q_jQ_j^*)(\hat{A}^{-1}\hat{B})q_j$.

By the induction hypothesis, $\hat{Q}_j = Q_jD$ is a basis generated by applying the Arnoldi process to the matrix $\hat{A}^{-1}\hat{B}$. The subsequent step of Arnoldi produces the vector $\hat{q}_{j+1}$ satisfying

$$(\hat{A}^{-1}\hat{B})\hat{Q}_j = \hat{Q}_jH_j + h_{j+1,j}\hat{q}_{j+1}e_j^T$$

or equivalently

$$(\hat{A}^{-1}\hat{B})Q_jD = Q_jDH_j + h_{j+1,j}\hat{q}_{j+1}e_j^T. \tag{2.61}$$

Therefore,

$$\begin{aligned} (I - Q_jQ_j^*)(\hat{A}^{-1}\hat{B})q_j &= \frac{1}{d_j}(I - Q_jQ_j^*)(\hat{A}^{-1}\hat{B})Q_jDe_j \\ &= \frac{1}{d_j}(I - Q_jQ_j^*)(h_{j+1,j}\hat{q}_{j+1}) \\ &= \left(\frac{h_{j+1,j}}{d_j}\right)\hat{q}_{j+1}. \end{aligned}$$

Thus q_{j+1} and $\hat{q}_{j+1}$ must be scalar multiples of one another.

2. The proof is similar to that of part 1. Let Q_j be the basis produced by the first

j iterations of Algorithm (2.8) with $(\tilde{\mu}_{j+1}, \tilde{u}_{j+1}) = (0, e_j)$. The residual is then

$$\begin{aligned} r &= Q_{j+1}^*(\hat{A}^{-1}\hat{C})Q_{j+1}e_j \\ &= \begin{pmatrix} Q_j^*(\hat{A}^{-1}\hat{C})q_j \\ q_{j+1}^*(\hat{A}^{-1}\hat{C})q_j \end{pmatrix}. \end{aligned}$$

Therefore, up to scale, q_{j+1} equals the projection onto the orthogonal complement of Q_j

$$(I - Q_jQ_j^*)(\hat{A}^{-1}\hat{C})q_j. \tag{2.62}$$

Letting $\hat{Q}_j$ be the orthonormal basis generated by applying the Arnoldi process to $\hat{A}^{-1}\hat{C}$ with starting vector q_1, we have $\hat{Q}_j = Q_jD$ for some diagonal matrix D by the induction hypothesis. The next vector $\hat{q}_{j+1}$ produced by the Arnoldi process satisfies

$$(\hat{A}^{-1}\hat{C})\hat{Q}_j = \hat{Q}_jH_j + h_{j+1,j}\hat{q}_{j+1}e_j^T$$

or equivalently

$$(\hat{A}^{-1}\hat{C})Q_jD = Q_jDH_j + h_{j+1,j}\hat{q}_{j+1}e_j^T. \tag{2.63}$$

Therefore we have

$$\begin{aligned} (I - Q_jQ_j^*)(\hat{A}^{-1}\hat{B})q_j &= \frac{1}{d_j}(I - Q_jQ_j^*)(\hat{A}^{-1}\hat{B})Q_jDe_j \\ &= \frac{1}{d_j}(I - Q_jQ_j^*)(h_{j+1,j}\hat{q}_{j+1}) \\ &= \left(\frac{h_{j+1,j}}{d_j}\right)\hat{q}_{j+1} \end{aligned}$$

and so $\hat{q}_{j+1}$ and q_{j+1} must be scalar multiples of one another.

3. Let $\hat{Q}_j$ be a basis generated by the Arnoldi-type Krylov process in Algorithm 2.2. Recall that the process is defined by the recurrences in Equations (2.5, 2.6). It

follows that the next basis vector $\hat{q}_{j+1}$ satisfies

$$\begin{aligned}(h_{C;j+1,j/2})\hat{q}_{j+1} &= (I - \hat{Q}_j\hat{Q}_j^*)\hat{A}^{-1}\hat{C}\hat{q}_{j/2}, \quad j \text{ even}\\ (h_{B;j+1,\lceil j/2\rceil})\hat{q}_{j+1} &= (I - \hat{Q}_j\hat{Q}_j^*)\hat{A}^{-1}\hat{B}\hat{q}_{\lceil j/2\rceil}, \quad j \text{ odd}.\end{aligned} \tag{2.64}$$

Next, let Q_j be the orthonormal basis produced by the first j iterations of Algorithm (2.8). By the induction hypothesis, $\hat{Q}_j = Q_jD$. Both bases are orthonormal; therefore, $D^*D = D^*Q_j^*Q_jD = \hat{Q}_j^*\hat{Q}_j = I$ and $(I - Q_jQ_j^*) = (I - \hat{Q}_j\hat{Q}_j^*)$. For odd j, Algorithm (2.2) generates a vector q_{j+1} which is a scalar multiple of $(I - Q_jQ_j^*)\hat{A}^{-1}\hat{B}q_{\lceil j/2\rceil}$ (by the argument in the proof of part 1). For even j, we have $(\tilde{\mu}_{j+1}, \tilde{u}_{j+1}) = (0, e_{j/2})$, and therefore Algorithm (2.2) generates a vector q_{j+1} which is a scalar multiple of $(I - Q_jQ_j^*)\hat{A}^{-1}\hat{C}q_{j/2}$, as in the proof of part 2. Thus, q_{j+1} and $\hat{q}_{j+1}$ must agree up to scale for all j.

□

At this point, we observe the following. Suppose Algorithm (2.8) is run with approximate solution $(\tilde{\mu}_{j+1}, \tilde{u}_{j+1})$, where the approximate eigenvector $\tilde{u}_{j+1}$ is partitioned into the leading j entries $\tilde{u}_{j+1;1:j}$ and the $(j+1)$-st entry $\tilde{u}_{j+1;j+1}$. The residual can be written

$$\begin{aligned} r &= [Q_j, q_{j+1}]^*(\tilde{\mu}^2 I + \tilde{\mu}\hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C})[Q_j, q_{j+1}]\tilde{u}\\ &= \begin{pmatrix} Q_j^*(\tilde{\mu}^2 I + \tilde{\mu}\hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C})[Q_j, q_{j+1}]\tilde{u} \\ q_{j+1}^*(\tilde{\mu}^2 I + \tilde{\mu}\hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C})[Q_j, q_{j+1}]\tilde{u}\end{pmatrix}\\ &= \begin{pmatrix} Q_j^*(\tilde{\mu}^2 I + \tilde{\mu}\hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C})Q_j\tilde{u}_{j+1;1:j} \\ q_{j+1}^*(\tilde{\mu}^2 I + \tilde{\mu}\hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C})Q_j\tilde{u}_{j+1;1:j}\end{pmatrix}\\ &\quad + \tilde{u}_{j+1;j+1}\begin{pmatrix} Q_j^*(\tilde{\mu}\hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C})q_{j+1} \\ q_{j+1}^*(\tilde{\mu}^2 I + \tilde{\mu}\hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C})q_{j+1}\end{pmatrix}. \end{aligned}$$

If the $(j+1)$-st entry $\tilde{u}_{j+1;j+1}$ is zero (as is the case for all choices of approximate eigenpair discussed in Theorem 2.5.1), then q_{j+1} is just the normalized projection of the residual $(\tilde{\mu}^2 I + \tilde{\mu}\hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C})(Q_j\tilde{u}_{j+1;1:j})$ onto the orthogonal complement of Q_j. The subspace spanned by Q_{j+1} is thus

$$\begin{aligned}\operatorname{span}\{Q_{j+1}\} &= \operatorname{span}\{Q_j, (\tilde{\mu}^2 I + \tilde{\mu}\hat{A}^{-1}\hat{B} + \hat{A}^{-1}\hat{C})(Q_j\tilde{u}_{j+1;1:j})\} \\ &= \operatorname{span}\{Q_j, \hat{A}^{-1}(\tilde{\mu}^2\hat{A} + \tilde{\mu}\hat{B} + \hat{C})(Q_j\tilde{u}_{j+1;1:j})\}.\end{aligned}$$

In other words, the subspace is enlarged by the following Cayley transform:

$$(\sigma^2 A + \sigma B + C)^{-1}(\tilde{\lambda}^2 A + \tilde{\lambda} B + C)(Q_j\tilde{u}_{j+1;1:j})$$

where $\tilde{\lambda} = \sigma + 1/\tilde{\mu}$.

Chapter 3

Projections based on moment matching

3.1 Moment matching of Villemagne and Skelton

Next, we will discuss a connection between generalized eigenvalue problems and linear transfer functions. A *linear transfer function* is a matrix-valued function

$$H(s) = L^*(G + sC)^{-1}B$$

of a complex scalar s, where $C, G \in \mathbb{C}^{n\times n}$ are square and $L \in \mathbb{C}^{n\times p}$, $B \in \mathbb{C}^{n\times m}$ are complex matrices of consistent sizes. Assuming G is nonsingular, the transfer function can be written in series form as

$$H(s) = \sum_{i=0}^{\infty}(-1)^i s^i M_i$$

where $M_i = L^*(G^{-1}C)^i G^{-1}B$. The matrices $\{M_i\}$ are termed the *moments* of the transfer function $H(s)$. Given another transfer function $H_R(s) = L_R^*(G_R + sC_R)^{-1}B_R$ with moments $\{M_{Ri}\}$, let k be the largest integer such that $M_i = M_{Ri}$ for all $i = 1, 2, \ldots, k$ (i.e. the first k moments of each transfer function match). We can then conclude that $H(s) = H_R(s) + O(s^{k+1})$. The goal of the moment-matching approach to model reduction is to construct $H_R(s)$ of modest dimension so that as many of the moments match as possible.

One way to produce a reduced transfer function $H_R(s)$ from the original transfer function $H(s)$ is to construct L_R, B_R, G_R, C_R by applying projections to L, B, G, C, respectively. The number of matching moments in this case is given by the following theorem, proven originally for $G = I$ by Villemagne and Skelton [14] and proven in general by Grimme [23] and Li [33].

Theorem 3.1.1. *Choose $X, Y \in \mathbb{C}^{n\times m}$ such that Y^*GX is nonsingular. Let $L_R = X^*L$, $G_R = Y^*GX$, $C_R = Y^*CX$, $B_R = Y^*B$. If*

$$\mathcal{K}_q(G^{-1}C, G^{-1}B) \subseteq \operatorname{span}\{x_1, \dots, x_m\}$$

$$\mathcal{K}_r(G^{-*}C^*, G^{-*}L) \subseteq \operatorname{span}\{y_1, \dots, y_m\},$$

then the following hold:

$$X(G_R^{-1}C_R)^i G_R^{-1} B_R = (G^{-1}C)^i G^{-1}B, \qquad 0 \le i \le q-1 \tag{3.1}$$

$$L_R^* G_R^{-1}(C_R G_R^{-1})^j Y^* = L^* G^{-1}(CG^{-1})^j, \qquad 0 \le j \le r-1 \tag{3.2}$$

$$M_i = M_{Ri} \qquad 0 \le i \le q+r-1. \tag{3.3}$$

This theorem can be used to simplify convergence analysis for a variety of related algorithms. See Li [33] for discussion of the application to asymptotic waveform evaluation, Pade via Lanczos (PvL), PRIMA, and the Krylov-type method proposed by Su and Craig [53, 40, 19]. The following discussion uses this theorem to describe the eigenvalue convergence of a generalized reduced-order eigenvalue problem $C_R u = \mu G_R u$ to the full-scale generalized eigenproblem $Cx = \lambda Gx$.

Suppose that both $G^{-1}C$ and $G_R^{-1}C_R$ are diagonalizable, i.e. there exist nonsingular matrices $V, \tilde{V}$ and diagonal Λ, Ω so that

$$G^{-1}C = V\Lambda V^{-1}, \qquad G_R^{-1}C_R = \tilde{V}\Omega\tilde{V}^{-1}. \tag{3.4}$$

We further assume that $\Lambda = \begin{pmatrix} \Lambda_1 & \\ & \Lambda_2 \end{pmatrix}$ and $\Omega = \begin{pmatrix} \Omega_1 & \\ & \Omega_2 \end{pmatrix}$ are partitioned so that Ω_1 approximates the desired spectrum Λ_1, and Λ_2, Ω_2 are relatively far. Let q, r be as in Theorem 3.1.1, and choose a polynomial $\phi(x) = \sum \alpha_i x^i$ of degree less than $q + r$. By Equation (3.3) we have

$$\begin{aligned} L^*\phi(G^{-1}C)G^{-1}B &= \sum_{i=0}^{q+r-1} \alpha_i M_i \\ &= \sum_{i=0}^{q+r-1} \alpha_i M_{Ri} \\ &= L_R^*\phi(G_R^{-1}C_R)G_R^{-1}B_R \end{aligned}$$

for any rectangular L, B of suitable dimension and L_R, B_R defined as in Theorem 3.1.1. Substituting Equation (3.4) gives

$$L^*V\phi(\Lambda)V^{-1}G^{-1}B = L_R^*\tilde{V}\phi(\Omega)\tilde{V}^{-1}G_R^{-1}B_R. \tag{3.5}$$

Partitioning $V^{-1}G^{-1}B = \begin{pmatrix} W_1 \\ W_2 \end{pmatrix}$ and $V^*L = \begin{pmatrix} W_3 \\ W_4 \end{pmatrix}$, we have

$$W_3^*\phi(\Lambda_1)W_1 - \hat{W}_3^*\phi(\Omega_1)\hat{W}_1 = \hat{W}_4^*\phi(\Omega_2)\hat{W}_2 - W_4^*\phi(\Lambda_2)W_2 \tag{3.6}$$

with $\hat{V}^{-1}G_R^{-1}B_R = \begin{pmatrix} \hat{W}_1 \\ \hat{W}_2 \end{pmatrix}$, $\hat{V}^*L_R = \begin{pmatrix} \hat{W}_3 \\ \hat{W}_4 \end{pmatrix}$ defined similarly. Thus if ϕ is chosen so that ϕ is close to 1 at Λ_1, Ω_1 and small at Λ_2, Ω_2, then $W_3^*W_1 \approx \hat{W}_3^*\hat{W}_1$. From this, a connection between W_1 and $\hat{W}_1$ might be derived in terms of the behavior of ϕ.

3.2 Projection methods via linearization

In light of Theorem 3.1.1, we seek structure-preserving projections of the monic quadratic eigenvalue problem that can offer a higher order of convergence than the methods discussed in 2.2.1. In order to preserve symmetry, we construct an orthonormal basis Q_m; the full-scale problem $(\lambda^2 I - \lambda B - C)x = 0$ is approximated by the projected problem $(\mu^2 I_m - \mu B_m - C_m)u = 0$, where $B_m = Q_m^T B Q_m$ and $C_m = Q_m^T C Q_m$. Linearizing in the usual way gives an equivalent pair of eigenproblems of twice the dimension

$$\tilde{C} \begin{pmatrix} x \\ \lambda x \end{pmatrix} = \lambda \tilde{G} \begin{pmatrix} x \\ \lambda x \end{pmatrix}$$
$$\tilde{C}_R \begin{pmatrix} u \\ \mu u \end{pmatrix} = \mu \tilde{G}_R \begin{pmatrix} u \\ \mu u \end{pmatrix}$$

where

$$\tilde{C} = \begin{pmatrix} 0 & I \\ C & B \end{pmatrix} \qquad \tilde{G} = I \tag{3.7}$$
$$\tilde{C}_R = \begin{pmatrix} 0 & I_m \\ C_m & B_m \end{pmatrix} \qquad \tilde{G}_R = I. \tag{3.8}$$

In order to apply the theory from the previous section we need left and right bases Y_{2m}, X_{2m} so that $\tilde{G}_R = Y_{2m}^* \tilde{G} X_{2m}$ and $\tilde{C}_R = Y_{2m}^* \tilde{C} X_{2m}$. A natural pair of bases that accomplish this is $X_{2m} = Y_{2m} = \begin{pmatrix} Q_m & 0 \\ 0 & Q_m \end{pmatrix}$, where Q_m is orthonormal. Therefore, the problem reduces to choosing Q_m so that the columns of X_{2m}, i.e.

$$\operatorname{span}\left\{ \begin{pmatrix} q_1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ q_1 \end{pmatrix}, \begin{pmatrix} q_2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ q_2 \end{pmatrix}, \ldots, \begin{pmatrix} q_m \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ q_m \end{pmatrix} \right\}$$

span as much of the left and right Krylov spaces $\mathcal{K}(\tilde{C}^T, L)$ and $\mathcal{K}(\tilde{C}, B)$ as possible.

3.2.1 The SOAR algorithm

The first projection method we discuss with the help of Theorem 3.1.1 is called SOAR (second-order Arnoldi) by Bai and Su [6, 7]. Let us apply the Arnoldi method to $\tilde{C}$ with an appropriate partitioning to give the defining equation

$$\begin{pmatrix} 0 & I \\ C & B \end{pmatrix} \begin{pmatrix} V_m \\ W_m \end{pmatrix} = \begin{pmatrix} V_m \\ W_m \end{pmatrix} H_m + h_{m+1,m} \begin{pmatrix} v_{m+1} \\ w_{m+1} \end{pmatrix} e_m^T. \tag{3.9}$$

A Gram-Schmidt process is used to construct the vectors V_m, W_m, but with a special choice of inner product so that the computed W_m is orthonormal:

$$\begin{pmatrix} V_{m+1} \\ W_{m+1} \end{pmatrix}^T \begin{pmatrix} 0 & \\ & I \end{pmatrix} \begin{pmatrix} V_{m+1} \\ W_{m+1} \end{pmatrix} = I_{m+1}. \tag{3.10}$$

This inner product is of course semi-definite, which raises the issue of breakdown. In addition to the usual "good" breakdown, where the residual of the orthogonalization process is zero, there is now a possibility of breakdown when the computed w_{m+1} is zero but v_{m+1} is not. Since the residual cannot then be normalized, the recurrence fails. Unlike the good case, this breakdown does not indicate that any convergence has taken place. For the rest of this discussion, we assume that no breakdown occurs, good or bad.

To start the recurrence, we may begin with an initial vector of the form $\begin{pmatrix} v_1 \\ w_1 \end{pmatrix} = \begin{pmatrix} 0 \\ q \end{pmatrix}$, where q is any unit n-vector. The right Krylov space of $\tilde{C}$ is spanned by the set of m $2n$-vectors generated by Equation (3.9):

$$\begin{aligned} K = \mathcal{K}_m\left(\tilde{C}, \begin{pmatrix} 0 \\ q \end{pmatrix}\right) &= \operatorname{span}\left\{ \begin{pmatrix} v_1 \\ w_1 \end{pmatrix}, \ldots, \begin{pmatrix} v_m \\ w_m \end{pmatrix} \right\} \\ &\subseteq \operatorname{span}\left\{ \begin{pmatrix} v_1 \\ 0 \end{pmatrix}, \ldots, \begin{pmatrix} v_m \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ w_1 \end{pmatrix}, \ldots, \begin{pmatrix} 0 \\ w_m \end{pmatrix} \right\}. \end{aligned}$$

From Equation (3.9) we have $W_m = V_m H_m + h_{m+1,m} v_{m+1}$, and so $\operatorname{span}\{w_1, \ldots, w_m\} = \operatorname{span}\{v_1, \ldots, v_{m+1}\}$ by the special structure of our $\tilde{C}$. It follows that K is contained in the span of the columns of $\begin{pmatrix} W_m & \\ & W_m \end{pmatrix}$. Therefore, choosing $Q_m = W_m$ and projecting as in the previous section yields a method whose convergence is determined by polynomials of degree at least m. An outline of the algorithm appears in Figure 3.1.

```
Input: $\|q\|_2 = 1$, $m > 0$
$v_1 = 0$; $w_1 = q$
for $j = 1, 2, \ldots, m$ do
    $v = w_j$; $w = Cv_j + Bw_j$
    for $i = 1, 2, \ldots, j$ do
        $h_{ij} = w_i^T w$
        $v = v - h_{ij} v_i$; $w = w - h_{ij} w_i$
    end for
    $h_{j+1,j} = \|w\|$
    $v_{j+1} = v / h_{j+1,j}$; $w_{j+1} = w / h_{j+1,j}$
end for
Returns orthonormal basis $W_{m+1}$.
```

Figure 3.1: SOAR algorithm

3.2.2 The Q-Arnoldi algorithm

A related Arnoldi-based algorithm is the Q-Arnoldi algorithm of Meerbergen and Robbé [35]. Unlike SOAR, the Q-Arnoldi algorithm constructs an orthonormal basis $\begin{pmatrix} V_m \\ W_m \end{pmatrix}$, using the Arnoldi-type recurrence from Equation (3.9). It follows from the structure of the linearized quadratic problem that for all $j > 1$,

$$W_{j-1} = V_j H_{1:j,1:j-1}. \tag{3.11}$$

The algorithm uses this identity to eliminate storage of the vectors $w_1, \ldots, w_{m-1}$, reducing memory usage approximately by half. Then, the upper-triangular entries of H are given by

$$\begin{aligned} h_{ij} &= \begin{pmatrix} v_i \\ w_i \end{pmatrix}^T \begin{pmatrix} v \\ w \end{pmatrix} \\ &= v_i^T v + w_i^T w \\ &= \begin{cases} v_i^T v + (V_{i+1} H_{1:i+1,1:i} e_i)^T w, & i < j \\ v_j^T v + w_j^T w, & i = j \end{cases} \end{aligned} \tag{3.12}$$

where $\begin{pmatrix} v \\ w \end{pmatrix} = \tilde{C} \begin{pmatrix} v_j \\ w_j \end{pmatrix}$. The succeeding vector $\begin{pmatrix} v_{j+1} \\ w_{j+1} \end{pmatrix}$ is computed by normalizing the result of a modified Gram-Schmidt orthogonalization, using the coefficients

```
**Input:** $\|q\|_2 = 1$, $m > 0$
$v_1 = 0$; $w_1 = q$
**for** $j = 1, 2, \ldots, m$ **do**
    $v = w_j$; $w = Cv_j + Bw_j$
    **for** $i = 1, 2, \ldots, j-1$ **do**
        $h_{ij} = v_i^T v + (\sum_{p=1}^{i+1} h_{pi} v_p)^T w$
        $v = v - h_{ij} v_i$; $w = w - h_{ij} w_i$
    **end for**
    $h_{jj} = v_j^T v + w_j^T w$
    $v = v - h_{jj} v_j$; $w = w - h_{jj} w_j$
    $h_{j+1,j} = \sqrt{\|v\|^2 + \|w\|^2}$
    $v_{j+1} = v/h_{j+1,j}$; $w_{j+1} = w/h_{j+1,j}$
**end for**
Returns basis $V_{m+1}$.
```

Figure 3.2: Q-Arnoldi algorithm

from (3.12).

The completed Q-Arnoldi algorithm appears in Figure 3.2. The next section makes further use of symmetry in the original quadratic eigenvalue problem to construct a Lanczos recurrence.

3.2.3 A generalized Lanczos variation of SOAR

The Arnoldi implementation used in SOAR can be applied to any monic quadratic eigenvalue problem $(\lambda^2 I - \lambda B - C)x = 0$, regardless of any special properties which B, C may possess. For the special case when B, C are both symmetric, then a change of inner product takes advantage of the symmetry to produce a Lanczos recurrence instead of Arnoldi. As in SOAR, we generate a Krylov space of the linearization $\tilde{C} = \begin{pmatrix} 0 & I \\ C & B \end{pmatrix}$. Since B, C are symmetric, the product $\begin{pmatrix} C & \\ & I \end{pmatrix} \tilde{C}$ is also symmetric. Therefore, we might attempt to construct a symmetric Lanczos recurrence with respect to the inner product $\begin{pmatrix} C & \\ & I \end{pmatrix}$:

$$\begin{pmatrix} 0 & I \\ C & B \end{pmatrix} \begin{pmatrix} V_m \\ W_m \end{pmatrix} = \begin{pmatrix} V_m \\ W_m \end{pmatrix} T_m + \beta_{m+1} \begin{pmatrix} v_{m+1} \\ w_{m+1} \end{pmatrix} e_m^T, \tag{3.13}$$

$$\begin{pmatrix} V_{m+1} \\ W_{m+1} \end{pmatrix}^T \begin{pmatrix} C & \\ & I \end{pmatrix} \begin{pmatrix} V_{m+1} \\ W_{m+1} \end{pmatrix} = I_{m+1}. \tag{3.14}$$

The difficulty is that our inner product may be indefinite, as in the case of SOAR. This has two side effects. One is the possibility of a "bad" breakdown when the computed v_{m+1}, w_{m+1} are quasi-null; as before, we assume that this will not happen. The other effect is that unless C is positive definite or semi-definite, we could have $v_{m+1}^T C v_{m+1} + w_{m+1}^T w_{m+1} < 0$ regardless of scaling. Indeed, observe that if m steps of the Lanczos recurrence in (3.13–3.14) can be performed without breakdown, then T_m is also the matrix generated by a nonsymmetric Lanczos recurrence; nonsymmetric Lanczos is not likely to produce a symmetric projection of the original nonsymmetric problem for many iterations. The eigenvalue problem $\tilde{C}\tilde{x} = \lambda \tilde{x}$ can be restated as the symmetric indefinite generalized eigenvalue problem

$$\begin{pmatrix} 0 & C \\ C & B \end{pmatrix} x = \lambda \begin{pmatrix} C & \\ & I \end{pmatrix} x$$

which suggests the symmetric indefinite Lanczos recurrence [31, 42]

$$\begin{pmatrix} 0 & I \\ C & B \end{pmatrix} \begin{pmatrix} V_m \\ W_m \end{pmatrix} = \begin{pmatrix} V_m \\ W_m \end{pmatrix} D_m^{-1} T_m + \frac{\beta_{m+1}}{d_{m+1}} \begin{pmatrix} v_{m+1} \\ w_{m+1} \end{pmatrix} e_m^T, \tag{3.15}$$

$$\begin{pmatrix} V_{m+1} \\ W_{m+1} \end{pmatrix}^T \begin{pmatrix} C & \\ & I \end{pmatrix} \begin{pmatrix} V_{m+1} \\ W_{m+1} \end{pmatrix} = D_{m+1} \tag{3.16}$$

where $D_m = \operatorname{diag}\{d_1, \dots, d_m\}$ is a diagonal matrix, possibly indefinite. We assume that each computed d_i is nonzero, else breakdown occurs.

The rest of the construction of the algorithm is similar to SOAR. Starting with

```
**Input:** $\|q_1\|_2 = 1$, $m > 0$
$v_1 = 0$; $w_1 = q_1$; $z = 0$; $\beta_1 = 0$; $d_0 = d_1 = 1$
for $j = 1, 2, \ldots, m$ do
    $v = w_j$; $w = z + Bw_j$
    $v = v - d_{j-1}\beta_j v_{j-1}$; $w = w - d_{j-1}\beta_j w_{j-1}$
    $\alpha_j = z^T v + w_j^T w$
    $v = v - \alpha_j v_j$; $w = w - \alpha_j w_j$
    $z = Cv$
    $d_{j+1} = \text{sign}(v^T z + w^T w)$
    $\beta_{j+1} = \sqrt{|v^T z + w^T w|}$
    $v_{j+1} = d_{j+1} v / \beta_{j+1}$; $w_{j+1} = d_{j+1} w / \beta_{j+1}$
    $z = d_{j+1} z / \beta_{j+1}$
    $q = w_{j+1}$ {One step of QR factorization}
    for $i = 1, 2, \ldots, j$ do
        $r_{i,j+1} = q_i^T q$
        $q = q - r_{i,j+1} q_i$
    end for
    $q_{j+1} = q / \|q\|$
end for
Returns orthonormal basis $Q_{m+1}$.
```

Figure 3.3: Lanczos-type SOAR algorithm

an initial vector of the form $\begin{pmatrix} 0 \\ q \end{pmatrix}$, the recurrence gives

$$\mathcal{K}_m\left(\tilde{C}, \begin{pmatrix} 0 \\ q \end{pmatrix}\right) \subseteq \text{span}\left\{\begin{pmatrix} v_1 \\ 0 \end{pmatrix}, \ldots, \begin{pmatrix} v_m \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ w_1 \end{pmatrix}, \ldots, \begin{pmatrix} 0 \\ w_m \end{pmatrix}\right\}$$

which in turn is spanned by the columns of $\begin{pmatrix} W_m & \\ & W_m \end{pmatrix}$. Here, W_m does not form an orthonormal basis. Therefore, we update the QR factorization $W_j = Q_j R_j$ as each vector w_j is computed. This orthogonalization step is approximately half as expensive as the orthogonalization in SOAR, since the vectors are of length n instead of $2n$.

Since the recurrence is short, at any step j we only need the two previous pairs of vectors $\{v_{j-1}, w_{j-1}\}$ and $\{v_j, w_j\}$ to compute v_{j+1}, w_{j+1}. A straightforward implementation of Lanczos would require additional matrix-vector multiplies when computing α_j and β_j, but these are eliminated by introducing another n-vector $z = Cv_j$ and ordering the operations carefully. Thus, our implementation (Figure 3.3) requires only an additional seven n-vectors beyond the $(m+1)$ vectors storing the generated basis Q_m. This is the major advantage over SOAR, which requires all of V_{m+1}, W_{m+1} at step m. The required number of gaxpys (vector computations of the form $\alpha x + \beta y$) is also approximately halved. Other costs are comparable, as summarized in the following table.

Cost	SOAR algorithm	Lanczos-type SOAR
Memory usage (n-vectors)	$2m+4$	$m+8$
Matrix-vector multiplies	$2m$	$2m$
$x \longleftarrow x + \alpha y$	$m(m+1)$	$4m + \frac{m(m+1)}{2}$
Dot products $v^T w$	$\frac{m(m+1)}{2}$	$4m + \frac{m(m+1)}{2}$

3.3 Reduction via nonsymmetric Lanczos

In the previous sections, three Arnoldi algorithms (SOAR, Q-Arnoldi, and Lanczos-type SOAR) were applied to the linearization $\tilde{C} = \begin{pmatrix} 0 & I \\ C & B \end{pmatrix}$. In each algorithm, m steps of Arnoldi generate a block diagonal orthonormal matrix X_{2m} whose columns span $\mathcal{K}_m(\tilde{C}, x_1)$; then the projection $X_{2m}^T \tilde{C} X_{2m}$ is a linearization of an order m symmetric quadratic eigenvalue problem. However, in general the space spanned by X_{2m} need not contain the left Krylov space. Therefore, not more than m moments of this projected quadratic eigenvalue problem must match those of the original (Theorem 3.1.1). In this section, we construct projected quadratic eigenvalue problems with a larger number of matching moments, using bases of both the left and right Krylov spaces of $\tilde{C}$.

A natural choice of method for constructing such bases iteratively is the nonsymmetric Lanczos recurrence. After applying $2m$ steps of nonsymmetric Lanczos to $\tilde{C}$, we obtain biorthogonal bases U_{2m}, V_{2m} so that

$$U_{2m}^T \begin{pmatrix} 0 & I \\ C & B \end{pmatrix} V_{2m} = T_{2m}. \tag{3.17}$$

Since U_{2m}, V_{2m} are bases of order $2m$ Krylov spaces of $\tilde{C}$ and $\tilde{C}^T$ respectively, the eigenvalues of the nonsymmetric tridiagonal T_{2m} converge according to polynomials of degree $4m$. It remains to be seen how to convert T_{2m} into an equivalent quadratic eigenvalue problem of order m. Label the entries of T_{2m} as follows:

$$T_{2m} = \begin{pmatrix} \alpha_1 & \gamma_2 & & \\ \beta_2 & \alpha_2 & \ddots & \\ & \ddots & \ddots & \gamma_{2m} \\ & & \beta_{2m} & \alpha_{2m} \end{pmatrix}.$$

By applying a shuffle $P = (e_1, e_3, \ldots, e_{2m-1}, e_2, \ldots, e_{2m})$ to the rows and columns of T_{2m}, we obtain the following block structure

$$P^T T_{2m} P = \left(\begin{array}{cccc|cccc} \alpha_1 & & & & \gamma_2 & & & \\ & \alpha_3 & & & \beta_3 & \gamma_4 & & \\ & & \ddots & & & \ddots & \ddots & \\ & & & \alpha_{2m-1} & & & \beta_{2m-1} & \gamma_{2m} \\ \hline \beta_2 & \gamma_3 & & & \alpha_2 & & & \\ & \beta_4 & \ddots & & & \alpha_4 & & \\ & & \ddots & \gamma_{2m-1} & & & \ddots & \\ & & & \beta_{2m} & & & & \alpha_{2m} \end{array}\right). \tag{3.18}$$

Let D_1, L, U, D_2 be the submatrices in Equation (3.18) labelled so that $P^T T_{2m} P = \begin{pmatrix} D_1 & L \\ U & D_2 \end{pmatrix}$. If breakdown did not occur in the Lanczos recurrence, then the bidiagonal matrices L and U are nonsingular. By letting $W = \begin{pmatrix} I & 0 \\ D_1 & L \end{pmatrix}$ and applying the similarity transformation

$$W \begin{pmatrix} D_1 & L \\ U & D_2 \end{pmatrix} W^{-1} = \begin{pmatrix} 0 & I \\ LU - LD_2L^{-1}D_1 & D_1 + LD_2L^{-1} \end{pmatrix}$$

we obtain a linearization of the following quadratic eigenvalue problem of order m:

$$[\mu^2 I - \mu(D_1 + LD_2L^{-1}) - (LU - LD_2L^{-1}D_1)]u = 0. \tag{3.19}$$

Likewise, each of the following three quadratic eigenvalue problems can be constructed from T_{2m} by similarity transformations:

$$[\mu^2 I - \mu(L^{-1}D_1L + D_2) - (UL - D_2L^{-1}D_1L)]u = 0 \tag{3.20}$$

$$[\mu^2 I - \mu(D_2 + UD_1U^{-1}) - (UL - UD_1U^{-1}D_2)]u = 0 \tag{3.21}$$

$$[\mu^2 I - \mu(U^{-1}D_2U + D_1) - (LU - D_1U^{-1}D_2U)]u = 0. \tag{3.22}$$

Thus the quadratic eigenvalue problems in Equations (3.19)-(3.22) have the same spectrum as T_{2m}, and therefore their eigenvalues converge as polynomials of degree $4m$; this is the best convergence result obtainable with Theorem 3.1.1. Unfortunately, if the original quadratic eigenvalue problem was symmetric, then these formulations of the projection would lose the original symmetry. In an application where the original QEP was nonsymmetric, this would not be considered a disadvantage. Note that the projected problems (3.19)-(3.22) do possess some structure: namely, the middle term is triangular and the constant term is Hessenberg (this is reminiscent of reducing a single matrix to its Schur form).

Next, we demonstrate how to update the reconstructed quadratic eigenvalue problem from (3.19) while running the nonsymmetric Lanczos recurrence. Suppose that the order m reduction has been completed; i.e. $2m$ steps of nonsymmetric Lanczos produced T_{2m}, from which the order m matrices $B_m = D_1 + LD_2L^{-1}$ and $C_m = LU - LD_2L^{-1}D_1$ were computed. Perform two more steps of nonsymmetric Lanczos to compute the additional six entries of T_{2m+2}:

$$T_{2m+2} = \left(\begin{array}{c|cc} T_{2m} & & \\ & \gamma_{2m+1} & \\ \hline \beta_{2m+1} & \alpha_{2m+1} & \gamma_{2m+2} \\ & \beta_{2m+2} & \alpha_{2m+2} \end{array}\right). \tag{3.23}$$

Permute T_{2m+2} as in (3.18), and label the resulting order $m+1$ submatrices $\hat{D}_1$, $\hat{D}_2$, $\hat{L}$, and $\hat{U}$. These matrices are readily obtained by extending D_1, D_2, L, and U, respectively:

$$\hat{D}_1 = \begin{pmatrix} D_1 & \\ & \alpha_{2m+1} \end{pmatrix} \qquad \hat{L} = \begin{pmatrix} L & \\ \beta_{2m+1}e_m^T & \gamma_{2m+2} \end{pmatrix} \tag{3.24}$$

$$\hat{D}_2 = \begin{pmatrix} D_2 & \\ & \alpha_{2m+2} \end{pmatrix} \qquad \hat{U} = \begin{pmatrix} U & \gamma_{2m+1}e_m \\ & \beta_{2m+2} \end{pmatrix}. \tag{3.25}$$

By a straightforward calculation, we have $\hat{L}^{-1} = \begin{pmatrix} L^{-1} & \\ x^T & 1/\gamma_{2m+2} \end{pmatrix}$ where the vector $x^T = (-\beta_{2m+1}/\gamma_{2m+2})e_m^T L^{-1}$. Consequently,

$$\hat{L}\hat{D}_2\hat{L}^{-1} = \begin{pmatrix} LD_2L^{-1} & \\ z^T & \alpha_{2m+2} \end{pmatrix}, \tag{3.26}$$

where $z^T = \beta_{2m+1}(\alpha_{2m} + \alpha_{2m+2})e_m^T L^{-1}$. Also, the product $\hat{L}\hat{U}$ is

$$\hat{L}\hat{U} = \begin{pmatrix} LU & \gamma_{2m}\gamma_{2m+1}e_m \\ \beta_{2m}\beta_{2m+1}e_m^T & \beta_{2m+1}\gamma_{2m+1} + \beta_{2m+2}\gamma_{2m+2} \end{pmatrix}, \tag{3.27}$$

and so the updated lower triangular/lower Hessenberg matrix pair is

$$\begin{aligned} B_{m+1} &= \hat{D}_1 + \hat{L}\hat{D}_2\hat{L}^{-1} \\ &= \begin{pmatrix} B_m & \\ z^T & \alpha_{2m+1} + \alpha_{2m+2} \end{pmatrix} \end{aligned} \tag{3.28}$$

$$\begin{aligned} C_{m+1} &= \hat{L}\hat{U} - \hat{L}\hat{D}_2\hat{L}^{-1}\hat{D}_1 \\ &= \begin{pmatrix} C_m & \gamma_{2m}\gamma_{2m+1}e_m \\ \beta_{2m}\beta_{2m+1}e_m^T - z^T D_1 & \zeta \end{pmatrix} \end{aligned} \tag{3.29}$$

for

$$\zeta = \beta_{2m+1}\gamma_{2m+1} + \beta_{2m+2}\gamma_{2m+2} - \alpha_{2m+1}\alpha_{2m+2},$$

$$z = \beta_{2m+1}(\alpha_{2m} + \alpha_{2m+2})L^{-T}e_m.$$

Thus, each update B_m, C_m to compute B_{m+1}, C_{m+1} requires one linear solve of a bidiagonal matrix and one diagonal matrix-vector multiply, at an $O(m)$ cost. Since the corresponding pair of nonsymmetric Lanczos steps required for an update has $O(n)$ cost, the complete algorithm produces an order m reduced model in $O(mn+m^2)$ time (and $8m$ matrix-vector products with B, C). Also, the storage requirements are modest: the Lanczos recurrences require storage for six work vectors of length $2n$, and the matrices B_m, C_m contain $m^2 + 2m - 1$ nonzeros. In addition, we must keep L and D_1 at an additional cost of $3m - 1$; these matrices are composed of the odd-numbered α_i, β_i and the even-numbered γ_i (in other words, the odd-numbered rows of T). The last even-numbered row of T is saved, but for only one iteration. The complete algorithm is outlined in Figure 3.4.

3.4 Convergence analysis of projection methods

Here, we present some results [26] bounding the convergence of an extreme Ritz value in the Lanczos-type process (Figure 2.4). The basic approach follows that of Ye [56], where a convergence analysis of the nonsymmetric Lanczos method is developed based largely on structural properties of tridiagonal matrices. The argument proceeds in two parts. First, we give a bound on the distance between a Ritz value θ_1 of $\hat{A}$ approximating an eigenvalue λ_1 of A. This bound depends on the moment matching of A and $\hat{A}$; in particular, it requires the largest power M so that the $(1, 1)$ entries of A^M, $\hat{A}^M$ match. Secondly, we use the nonzero structure of the matrices generated by the algorithm to determine this moment matching result.

```
**Input:** $u_1, v_1$ s.t. $u_1^T v_1 = 1$, $m > 0$
{First nonsymmetric Lanczos step}
$x = \begin{pmatrix} 0 & C^T \\ I & B^T \end{pmatrix} x_1$
$y = \begin{pmatrix} 0 & I \\ C & B \end{pmatrix} y_1$
$\alpha_1 = x_1^T y$
$x = x - \alpha_1 x_1$
$y = y - \alpha_1 y_1$
$\gamma_2 = \sqrt{|x^T y|}; \beta_2 = x^T y / \gamma_2$
$x_2 = x/\gamma_2; y_2 = y/\beta_2$
$y = \begin{pmatrix} 0 & I \\ C & B \end{pmatrix} y_2$
$\alpha_2 = x_2^T y$
{Initialize matrices}
$B_m = [\alpha_1 + \alpha_2]$; $C_m = [\beta_2\gamma_2 - \alpha_1\alpha_2]$
$L = [\gamma_2]$; $D_1 = [\alpha_1]$
{Begin main loop}
**for** $j = 1, 2, \ldots, m-1$ **do**
    Perform two steps of nonsymmetric Lanczos:
    **for** $k = 2j, 2j+1$ **do**
        $x = \begin{pmatrix} 0 & C^T \\ I & B^T \end{pmatrix} x_k$
        $x = x - \beta_k x_{k-1} - \alpha_k x_k$
        $y = y - \gamma_k y_{k-1} - \alpha_k y_k$
        $\gamma_{k+1} = \sqrt{|x^T y|}; \beta_{k+1} = x^T y / \gamma_{k+1}$
        $x_{k+1} = x/\gamma_{k+1}; y_{k+1} = y/\beta_{k+1}$
        $y = \begin{pmatrix} 0 & I \\ C & B \end{pmatrix} y_{k+1}$
        $\alpha_{k+1} = x_{k+1}^T y$
    **end for**
    {Update $B_m$, $C_m$, $L$, $D_1$}
    Solve $L^T z = \beta_{2j+1}(\alpha_{2j} + \alpha_{2j+2}) e_j$ for $z$
    $\zeta = \beta_{2j+1}\gamma_{2j+1} + \beta_{2j+2}\gamma_{2j+2} - \alpha_{2j+1}\alpha_{2j+2}$
    Update $B_m = \begin{pmatrix} B_m & \\ z^T & \alpha_{2j+1} + \alpha_{2j+2} \end{pmatrix}$
    Update $C_m = \begin{pmatrix} C_m & \gamma_{2j}\gamma_{2j+1} e_j \\ \beta_{2j}\beta_{2j+1} e_m^T - z^T D_1 & \zeta \end{pmatrix}$
**end for**
Returns order $m$ reduced model $\lambda^2 I - \lambda B_m - C_m$.
```

Figure 3.4: Tridiagonal-Hessenberg reduction via nonsymmetric Lanczos

3.4.1 Convergence of nonsymmetric Lanczos

We consider the nonsymmetric Lanczos method as an example. The following result on Ritz value convergence appears in Ye's paper [56], but is not specific to Lanczos. For simplicity, assume that both the original $n \times n$ matrix $A = Y^{-1}\Lambda Y$ and the $m \times m$ reduced order matrix $\hat{A} = Q^{-1}\Theta Q$ are diagonalizable. Write $P = Q^{-T}$, $X = Y^{-T}$. Suppose that we can find M so that

$$e_1^T A^i e_1 = e_1^T \hat{A}^i e_1 \tag{3.30}$$

for any $i \leq M$. Then for any polynomial f of degree less than or equal to M,

$$\begin{gathered} e_1^T f(T_n) e_1 = e_1^T f(T_m) e_1 \\ e_1^T X^T f(\Lambda) Y e_1 = e_1^T P^T f(\Theta) Q e_1 \\ f(\lambda_1) x_{11} y_{11} + \sum_{i \neq 1} f(\lambda_i) x_{i1} y_{i1} = f(\theta_1) p_{11} q_{11} + \sum_{i \neq 1} f(\theta_i) p_{i1} q_{i1} \end{gathered}$$

In particular, choose $f(x) = (x - \theta_1)\phi(x)$, where $\phi(x)$ is a polynomial of degree $M - 1$ with $\phi(\lambda_1) = 1$. Then

$$\begin{aligned} \lambda_1 - \theta_1 &= \frac{1}{x_{11} y_{11}} \left(\sum_{i \neq 1} (\theta_i - \theta_1)\phi(\theta_i) p_{i1} q_{i1} - \sum_{i \neq 1} (\lambda_i - \theta_1)\phi(\lambda_i) x_{i1} y_{i1} \right) \\ |\lambda_1 - \theta_1| &\leq \frac{K}{|x_{11} y_{11}|} \left(\sum_{i \neq 1} |\phi(\theta_i) p_{i1} q_{i1}| + \sum_{i \neq 1} |\phi(\lambda_i) x_{i1} y_{i1}| \right) \end{aligned} \tag{3.31}$$

where $K = \max_{i \neq 1} \{|\theta_i - \theta_1|, |\lambda_i - \theta_1|\}$. Thus, we have the following bound:

Theorem 3.4.1. *Suppose as above that $A = X^T \Lambda Y$ and $\hat{A} = P^T \Theta Q$ are diagonalizable ($P^T Q = I$, $X^T Y = I$). Let $\mathcal{S}$ be the set of $n - 1$ eigenvalues and $m - 1$ Ritz values excluding λ_1, θ_1. Also let $K = \max_{x \in \mathcal{S}} |x - \theta_1|$. If $M \geq 1$ is chosen so that*

$e_1^T A^i e_1 = e_1^T \hat{A}^i e_1$ *for all* $i \leq M$, *then*

$$\begin{aligned} |\lambda_1 - \theta_1| \leq K\epsilon & \frac{(\sum_{i\neq 1} |x_{i1}|^2 + \sum_{i\neq 1} |p_{i1}|^2)^{1/2}}{|x_{11}|} \\ & \cdot \frac{(\sum_{i\neq 1} |y_{i1}|^2 + \sum_{i\neq 1} |q_{i1}|^2)^{1/2}}{|y_{11}|} \end{aligned} \tag{3.32}$$

where $\epsilon = \min_{\substack{\phi\in\mathcal{P}_{M-1} \\ \phi(\lambda_1)=1}} \max_{x\in\mathcal{S}} |\phi(x)|$.

Next, we need the moment matching result. Assume that the nonsymmetric Lanczos recurrence on A can be run to termination, producing a tridiagonal $n \times n$ matrix T_n similar to A. Since T_m (obtained after m steps) is simply the order m principal submatrix of T_n, the following results are straightforward to show:

Theorem 3.4.2. *Let* T_m *be a principal submatrix of the tridiagonal matrix* T_n.

1. *For any* $i \leq m-1$, $T_n^i e_1 = \begin{pmatrix} T_m^i e_1 \\ 0 \end{pmatrix}$.

2. *For any* $i \leq 2m-1$, $e_1^T T_n^i e_1 = e_1^T T_m^i e_1$.

Therefore setting $M = 2m-1$ in Theorem 3.4.1 gives a convergence result for nonsymmetric Lanczos:

Corollary 3.4.3. *Suppose* $T_n = X^T \Lambda Y$ *and* $T_m = P^T \Theta Q$ *are diagonalizable* ($P^T Q = I$, $X^T Y = I$). *Let* $\mathcal{S}$ *be the set of* $n-1$ *eigenvalues and* $m-1$ *Ritz values excluding* λ_1, θ_1. *Then*

$$\begin{aligned} |\lambda_1 - \theta_1| \leq & \left(\max_{x\in\mathcal{S}} |x - \theta_1|\right) \left(\min_{\substack{\phi\in\mathcal{P}_{2m-2} \\ \phi(\lambda_1)=1}} \max_{x\in\mathcal{S}} |\phi(x)|\right) \\ & \cdot \left(\frac{(\sum_{i\neq 1} |x_{i1}|^2 + \sum_{i\neq 1} |p_{i1}|^2)^{1/2}}{|x_{11}|}\right) \left(\frac{(\sum_{i\neq 1} |y_{i1}|^2 + \sum_{i\neq 1} |q_{i1}|^2)^{1/2}}{|y_{11}|}\right) \end{aligned} \tag{3.33}$$

3.4.2 Extension to Lanczos-type process

The same approach works for the Lanczos-type process. Assuming that the process can run to completion, we have two $n \times n$ symmetric matrices T_B, T_C with the nonzero structure illustrated in Figure 2.3. To simplify the following, we characterize this type of structure by the following definition.

Definition 3.4.4. If $a_{ij} = 0$ whenever $i > mj + b_1$ or $j > mi + b_2$, then $A = (a_{ij})$ is called an (m, b_1, b_2)-*fan* matrix.

A tridiagonal matrix is a $(1, 1, 1)$-fan matrix, and T_B, T_C are $(2, 0, 0)$-fan and $(2, 1, 1)$-fan matrices respectively. Basic properties of fan matrices include the following:

Lemma 3.4.5. *Suppose A is an (m, b_1, b_2)-fan matrix and $\tilde{A}$ is $(\tilde{m}, \tilde{b}_1, \tilde{b}_2)$-fan. Then*

1. *A^T is (m, b_2, b_1)-fan.*

2. *$A + \tilde{A}$ is (m, b_1, b_2)-fan if $m \geq \tilde{m}$, $m + b_1 \geq \tilde{m} + \tilde{b}_1$, and $m + b_2 \geq \tilde{m} + \tilde{b}_2$.*

3. *The product $\tilde{A}A$ is $(m\tilde{m}, \tilde{m}b_1 + \tilde{b}_1, m\tilde{b}_2 + b_2)$-fan.*

Proof. The first two properties are immediate. The third follows by noting that

$$Ae_j \in \operatorname{span}\left\{e_k : k = 1 + \max\left(\left\lfloor \frac{j - b_2 - 1}{m} \right\rfloor, 0\right), \ldots, \min\left(mj + b_1, n\right)\right\}$$

and similarly

$$\tilde{A}^T e_i \in \operatorname{span}\left\{e_{\tilde{k}} : \tilde{k} = 1 + \max\left(\left\lfloor \frac{i - \tilde{b}_1 - 1}{\tilde{m}} \right\rfloor, 0\right), \ldots, \min\left(\tilde{m}i + \tilde{b}_2, n\right)\right\}.$$

The sets of indices are disjoint when $i > \tilde{m}(mj + b_1) + \tilde{b}_1$ or $j > m(\tilde{m}i + \tilde{b}_2) + b_2$. □

The following observation will be useful:

Lemma 3.4.6. *Let A be an (m, b_1, b_2)-fan matrix with principal $M \times M$ submatrix A_M. For any M-vector x, $A\begin{pmatrix} x \\ 0 \end{pmatrix} = \begin{pmatrix} A_M x \\ * \end{pmatrix}$. If $jm + b_1 \leq M$,*

$$Ae_j = \begin{pmatrix} A_M e_j \\ 0 \end{pmatrix}.$$

The original quadratic eigenvalue problem has the same eigenvalues as the linearization $\begin{pmatrix} 0 & I \\ T_C & T_B \end{pmatrix}$. The QEP resulting from an order m projection has the same eigenvalues as the corresponding linearization $\begin{pmatrix} 0 & I_m \\ (T_C)_m & (T_B)_m \end{pmatrix}$. We need an equivalent moment-matching result to Theorem 3.4.2 determining the maximum exponent M so that

$$e_1^T \begin{pmatrix} 0 & I \\ T_C & T_B \end{pmatrix}^i e_1 = e_1^T \begin{pmatrix} 0 & I_m \\ (T_C)_m & (T_B)_m \end{pmatrix}^i e_1$$

for all $i \leq M$; then the bound in Theorem 3.4.1 can be applied.

By induction, we can show that

$$\begin{pmatrix} 0 & I \\ T_C & T_B \end{pmatrix}^i = \begin{pmatrix} R_i & S_i \\ R_{i+1} & S_{i+1} \end{pmatrix}$$

defined by the recurrences

$$R_{i+1} = T_B R_i + T_C R_{i-1}, \qquad R_2 = T_C, \qquad R_1 = 0 \tag{3.34}$$

$$S_{i+1} = T_B S_i + T_C S_{i-1}, \qquad S_2 = T_B, \qquad S_1 = I. \tag{3.35}$$

Similarly, $\begin{pmatrix} 0 & I_m \\ (T_C)_m & (T_B)_m \end{pmatrix}^i = \begin{pmatrix} \hat{R}_i & \hat{S}_i \\ \hat{R}_{i+1} & \hat{S}_{i+1} \end{pmatrix}$ where $\hat{R}_i$ and $\hat{S}_i$ are defined by recurrences like (3.34, 3.35). This leads to the following theorem:

Theorem 3.4.7. *Let $T_B, T_C, (T_B)_m$, $(T_C)_m$ be defined as above. Also let $M = \lfloor \log_2 m \rfloor$, the largest integer such that $2^M \leq m$. Then for all $i \leq 2M+1$,*

$$e_1^T \begin{pmatrix} 0 & I \\ T_C & T_B \end{pmatrix}^i e_1 = e_1^T \begin{pmatrix} 0 & I_m \\ (T_C)_m & (T_B)_m \end{pmatrix}^i e_1.$$

Proof. Using Lemma 3.4.5 and recurrences (3.34), (3.35), it is not hard to show by induction that R_i is $(2^{i-1}, 2^{i-2}, 1)$-fan for $i \geq 2$, and S_i is $(2^{i-1}, 0, 0)$-fan for $i \geq 1$. Next, prove by induction that for $i \leq M$, $R_i e_1 = \begin{pmatrix} R_i e_1 \\ 0 \end{pmatrix}$; the inductive step is

$$\begin{aligned} R_i e_1 &= T_B R_{i-1} e_1 + T_C R_{i-2} e_1 \\ &= T_B \begin{pmatrix} \hat{R}_{i-1} e_1 \\ 0 \end{pmatrix} + T_C \begin{pmatrix} \hat{R}_{i-2} e_1 \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} (T_B)_m \hat{R}_{i-1} e_1 \\ 0 \end{pmatrix} + \begin{pmatrix} (T_C)_m \hat{R}_{i-2} e_1 \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} R_i e_1 \\ 0 \end{pmatrix}. \end{aligned}$$

It follows that

$$\begin{aligned} R_{M+1} e_1 &= T_B \begin{pmatrix} \hat{R}_M e_1 \\ 0 \end{pmatrix} + T_C \begin{pmatrix} \hat{R}_{M-1} e_1 \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} (T_B)_m \hat{R}_M e_1 \\ * \end{pmatrix} + \begin{pmatrix} (T_C)_m \hat{R}_{M-1} e_1 \\ * \end{pmatrix} \\ &= \begin{pmatrix} R_M e_1 \\ * \end{pmatrix}. \end{aligned}$$

Similarly, it can be shown that $e_1^T R_{i+1} = (e_1^T \hat{R}_{i+1}, *)$ and $e_1^T S_{i+1} = (e_1^T \hat{S}_{i+1}, 0)$. Therefore for $i, j \leq M$,

$$\begin{aligned} e_1^T \begin{pmatrix} O & I \\ T_C & T_B \end{pmatrix}^{i+1} &= (e_1^T R_{i+1}, e_1^T S_{i+1}) \\ &= (e_1^T \hat{R}_{i+1}, *, e_1^T \hat{S}_{i+1}, 0) \end{aligned} \tag{3.36}$$

and

$$\begin{aligned} \begin{pmatrix} O & I \\ T_C & T_B \end{pmatrix}^j e_1 &= \begin{pmatrix} R_j e_1 \\ R_{j+1} e_1 \end{pmatrix} \\ &= \begin{pmatrix} \hat{R}_j e_1 \\ 0 \\ \hat{R}_{j+1} e_1 \\ 0 \end{pmatrix}. \end{aligned} \tag{3.37}$$

The product of Equations (3.36, 3.37) is then

$$e_1^T \hat{R}_{i+1}\hat{R}_j e_1 + e_1^T \hat{S}_i \hat{R}_{j+1} e_1 = e_1^T \begin{pmatrix} 0 & I \\ (T_C)_m & (T_B)_m \end{pmatrix}^{i+j+1} e_1.$$

□

Therefore the following convergence bound applies to the Lanczos-type process.

Corollary 3.4.8. *Suppose the Lanczos-type algorithm in Figure 2.4 produces $(T_B)_m$, $(T_C)_m$ after m steps and can be run to termination to produce T_B, T_C. Suppose further that*

$$\begin{pmatrix} O & I \\ T_C & T_B \end{pmatrix} = X^T \Lambda Y, \quad \begin{pmatrix} O & I_m \\ (T_C)_m & (T_B)_m \end{pmatrix} = P^T \Theta Q$$

are diagonalizable ($P^TQ = I$, $X^TY = I$). Then

$$\begin{aligned} |\lambda_1 - \theta_1| \le \epsilon_{2M} \left(\max_{x \in \mathcal{S}} |x - \theta_1| \right) \cdot \left(\frac{(\sum_{i \ne 1} |x_{i1}|^2 + \sum_{i \ne 1} |p_{i1}|^2)^{1/2}}{|x_{11}|} \right) \\ \cdot \left(\frac{(\sum_{i \ne 1} |y_{i1}|^2 + \sum_{i \ne 1} |q_{i1}|^2)^{1/2}}{|y_{11}|} \right) \end{aligned} \tag{3.38}$$

where $\epsilon_{2M} = \min \left\{ \max_{x \in \mathcal{S}} |\phi(x)| : \phi \in \mathcal{P}_{2\lfloor \log_2 m \rfloor}, \phi(\lambda) = 1 \right\}$ and $\mathcal{S}$ is the set of $2n-1$ eigenvalues and $2m-1$ Ritz values excluding λ_1, θ_1.

Remark 3.4.9. A similar calculation gives a corresponding result for the Arnoldi process (Figure 2.2) where the polynomial ϕ is of degree $\lfloor \log_2 m \rfloor$ rather than $2\lfloor \log_2 m \rfloor$.

Chapter 4

Numerical experiments

4.1 An example from dissipative acoustics

As an application, we consider the following problem from dissipative acoustics [9, 55]. Let $\Omega \subset \mathbb{R}^2$ be a rectangular cavity filled with an acoustic fluid (such as air), with one absorbing wall Γ_A and three reflecting walls Γ_R.

Let $P(x,t)$ and $U(x,t)$ be the acoustic pressure and the fluid displacement, respectively. Also let ρ be the density of the fluid, and c the speed at which the fluid conducts sound. Then the behavior of the fluid satisfies the equations

$$\rho \frac{\partial^2 U}{\partial t^2} + \nabla P = 0 \tag{4.1}$$

$$-\rho c^2 \operatorname{div} U = P \tag{4.2}$$

with boundary conditions

$$U \cdot \nu = 0 \qquad \text{on } \Gamma_R \tag{4.3}$$

$$\alpha U \cdot \nu + \beta \frac{\partial U}{\partial t} \cdot \nu = P \qquad \text{on } \Gamma_A \tag{4.4}$$

where the scalars α, β describe the impedance of the absorbing material. As in [9], we choose $\rho = 1$ kg/m^3, $c = 340$ m/s, $\alpha = 5 \times 10^4$ N/m^3, $\beta = 200$ N $\cdot$ s/m^3 for our model problem; this choice of α and β models a very viscous absorbing material.

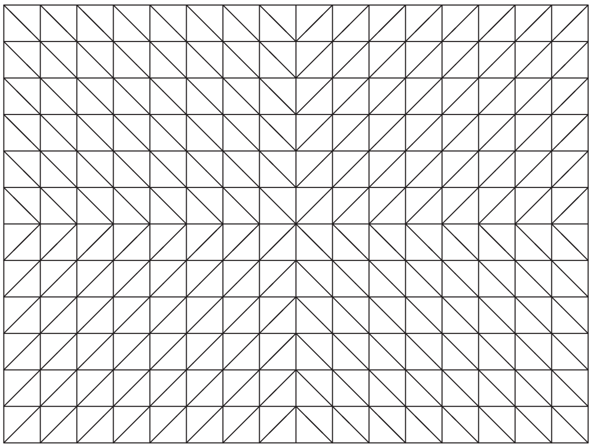


Figure 4.1: Triangulation of rectangular Ω, $N = 2$.

We are interested in finding the damped vibration modes of the fluid, which are solutions of the form $U(x,t) = e^{\lambda t}u(x)$, $P(x,t) = e^{\lambda t}p(x)$. Then, equations (4.1) – (4.4) reduce to finding λ, p, u satisfying

$$\begin{aligned} \rho\lambda^2 u + \nabla p &= 0 \qquad \text{in } \Omega \\ p &= -\rho c^2 \text{ div } u \qquad \text{in } \Omega \\ p &= (\alpha + \lambda\beta) u \cdot \nu \qquad \text{on } \Gamma_A \\ u \cdot \nu &= 0 \qquad \text{on } \Gamma_R. \end{aligned}$$

This system can be converted to a variational formulation. Let $\mathcal{V} = \{v \in H(\text{div}, \Omega) : v \cdot \nu \in L^2(\partial\Omega) \text{ and } v \cdot \nu = 0 \text{ on } \Gamma_R\}$. The problem is equivalent to finding $\lambda \in \mathbb{C}$, nonzero $u \in \mathcal{V}$ so that

$$\lambda^2 \int_\Omega \rho u \cdot v + \lambda \int_{\Gamma_A} \beta u \cdot \nu\, v \cdot \nu + \int_{\Gamma_A} \alpha u \cdot \nu\, v \cdot \nu + \int_\Omega \rho c^2 \text{ div } u \text{ div } v = 0 \tag{4.5}$$

for all $v \in \mathcal{V}$. Using finite elements to approximate $\mathcal{V}$ by $\mathcal{V}_h = \text{span}\{\phi_1, \ldots, \phi_n\}$ yields the $n \times n$ quadratic matrix eigenvalue problem

$$\lambda^2 M x + \lambda \beta F x + (\alpha F + K)x = 0, \tag{4.6}$$

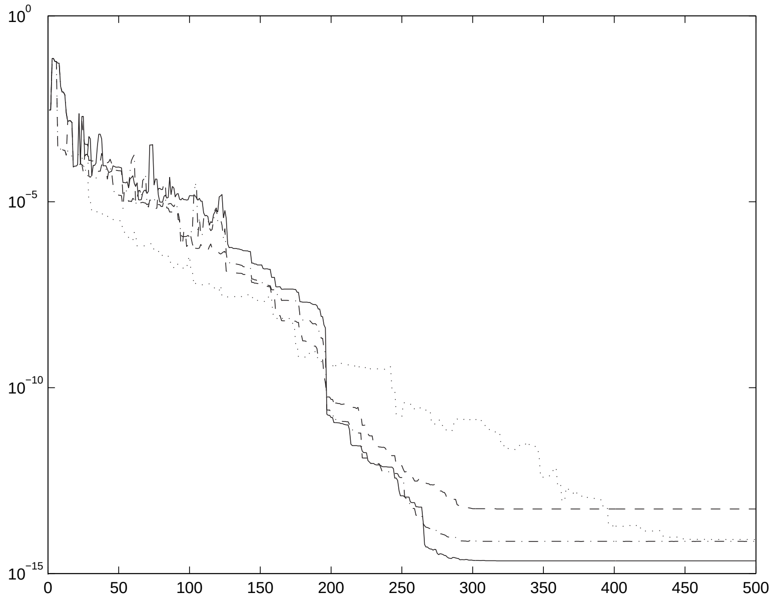


Figure 4.2: Relative residual norms for selected eigenvalues.

where

$$M_{ij} = \int_{\Omega} \rho \phi_i \cdot \phi_j, \; K_{ij} = \int_{\Omega} \rho c^2 \text{ div } \phi_i \text{ div } \phi_j, \; F_{ij} = \int_{\Gamma_A} \phi_i \cdot \nu \, \phi_j \cdot \nu.$$

To avoid spurious eigenvalues caused by discretization, lowest order Raviart-Thomas finite elements are used [9, 46]. Each basis element ϕ_i is a vector-valued function with piecewise constant divergence on each triangle of the mesh and $\phi_i \cdot \nu$ constant along each edge. With a natural choice of the basis, each finite element corresponds to an edge in the interior or on the absorbing boundary Γ_A. We use a triangulation of Ω with $6N$ edges along the vertical sides and $8N$ edges along the horizontal sides (Figure 4.1). With the choice of the discretization parameter $N = 8$, a model with 9168 degrees of freedom is obtained.

Let $A = M$, $B = \beta F$, $C = \alpha F + K$ and write (4.6) as the symmetric quadratic

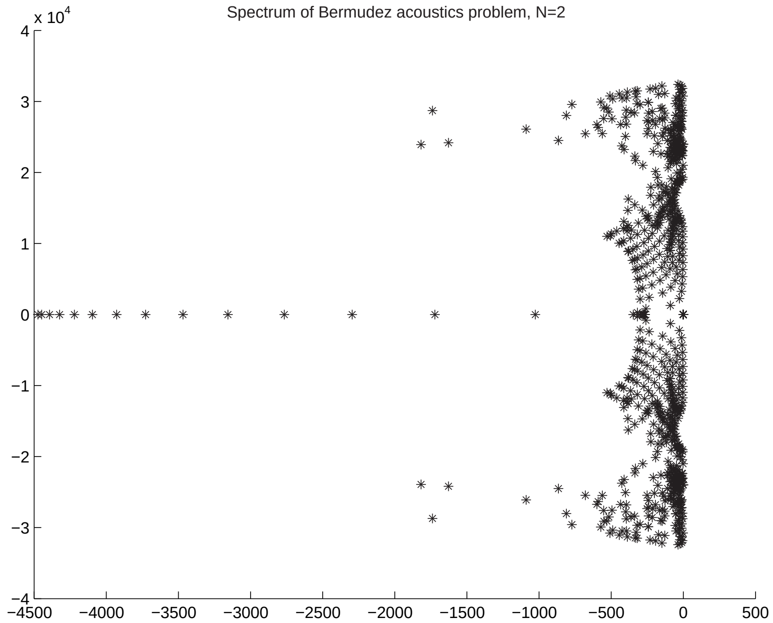


Figure 4.3: Spectrum of Bermúdez–Durán acoustics problem.

eigenvalue problem

$$(\lambda^2 A + \lambda B + C)x = 0 \tag{4.7}$$

where A is symmetric positive definite and B, C are positive semidefinite matrices. Observe that this problem has both *overdamped* (negative real) eigenvalues and complex eigenvalues appearing in conjugate pairs (Figure 4.3). Since this problem models a stable phenomenon, all eigenvalues appear in the left complex half-plane.

4.1.1 The symmetric Lanczos-type method with shift

Here we demonstrate the convergence behavior of the symmetric Lanczos-type method described in Section 2.2.1, by using it to compute the four eigenvalues of (4.7) with largest real part and nonnegative imaginary part. The acoustics problem is constructed on a rectangular domain with the choice of constants ρ, c, α, β given in Section 4.1. Also, the discretization parameter N is set to 8, giving a quadratic eigenvalue problem (4.7) of order 9168. In this test, we transform (4.7) by a real shift and invert, and apply the symmetric Lanczos-type method to the shifted problem. Shift and invert about $\sigma = -253$ to get

$$(\mu^2\hat{A} + \mu\hat{B} + \hat{C})x = 0;$$

for this choice of σ, $\hat{A}$ remains positive definite. Therefore, we can take the Cholesky decomposition $\hat{A} = LL^T$ and construct an equivalent monic problem

$$(\mu^2 I + \mu(L^{-1}\hat{B}L^{-T}) + (L^{-1}\hat{C}L^{-T}))u = 0 \tag{4.8}$$

where $\hat{A} = LL^T$ is the Cholesky decomposition. Algorithm 2.4 is applied to (4.8) in a symmetric Lanczos type process to get a basis V_k and banded $k \times k$ matrices T_a, T_b. It follows that if (θ_i, u_i) is an eigenpair to the projected problem

$$(\mu^2 I_i + \mu T_{a(1:i,1:i)} + T_{b(1:i,1:i)})u = 0, \tag{4.9}$$

then $(\lambda_i, x_i) = (\sigma + 1/\mu_i, z_i/\|z_i\|)$ is an approximate eigenpair to the original problem (4.7), where $z_i = L^T V_{k(:,1:i)} u_i$.

Figure 4.2 shows the convergence rates of Algorithm 2.4 when computing each of the four eigenvalues of (4.7) listed in Figure 4.4. Each line corresponds to one exact eigenvalue. For each subspace dimension $i = 1, \ldots, 500$, we perform i steps of Algorithm 2.4 to obtain (4.9), which is linearized as

$$\begin{pmatrix} & I_i \\ -T_{b(1:i,1:i)} & -T_{a(1:i,1:i)} \end{pmatrix} \begin{pmatrix} u \\ \mu u \end{pmatrix} = \mu \begin{pmatrix} u \\ \mu u \end{pmatrix}$$

and solved for the desired approximate eigenpair (λ_i, x_i) using the Matlab `eigs` function. To indicate the accuracy of this approximate solution, Figure 4.2 plots the resulting relative residual norms $r_i = \dfrac{\|(\lambda_i^2 A + \lambda_i B + C)x_i\|}{|\lambda_i|^2\|A\| + |\lambda_i|\|B\| + \|C\|}$.

Eigenvalue $\hat{\lambda}$	Plot line	Matrix-vector products	$\lvert\lambda_i - \hat{\lambda}\rvert$
$-259.23 + 813.27i$	dotted	318	1.746×10^{-8}
$-320.54 + 267.66i$	dashed	322	1.053×10^{-8}
-342.15	dash-dot	356	8.830×10^{-9}
-296.66	solid	386	3.797×10^{-9}

Figure 4.4: Required matrix-vector products for acoustics problem.

Figure 4.4 lists the values of the four selected eigenvalues. For each exact eigenvalue $\hat{\lambda}$, the table indicates which line in Figure 4.2 corresponds to that eigenvalue, the number of matrix-vector products required to obtain a relative residual norm $r_i < 10^{-8}$, and the accuracy of the corresponding eigenvalue λ_i.

4.1.2 Comparison of Arnoldi-type methods

In the following examples, we compare the convergence of the Arnoldi variant methods I-III when applied to the acoustics problem of order 564, obtained by setting the parameter $N = 2$. Each Arnoldi method is restarted periodically. For our tests, each method is run for a total of 50 iterations, restarting after every 10 iterations. In each test, we start with the shift $\sigma = -200 + 300i$ and a randomly-chosen initial vector; the closest eigenvalue to σ is $\lambda \approx -317.98 + 267.76i$. The accuracy of each Ritz pair (λ_i, x_i) is tracked by plotting its residual $\|(\lambda_i^2 A + \lambda_i B + C)x_i\|$.

In Figure 4.5, we show the convergence of each Arnoldi variant method when each restart shifts and inverts the quadratic eigenvalue problem about the current shift. Since the original problem is inverted, each inner Arnoldi implementation attempts to find a large, well-separated eigenvalue μ. Krylov methods converge quickly under these conditions, hence the rapid leveling-off of the residuals after each restart. However, the construction of each Arnoldi variant algorithm neglects one of the three

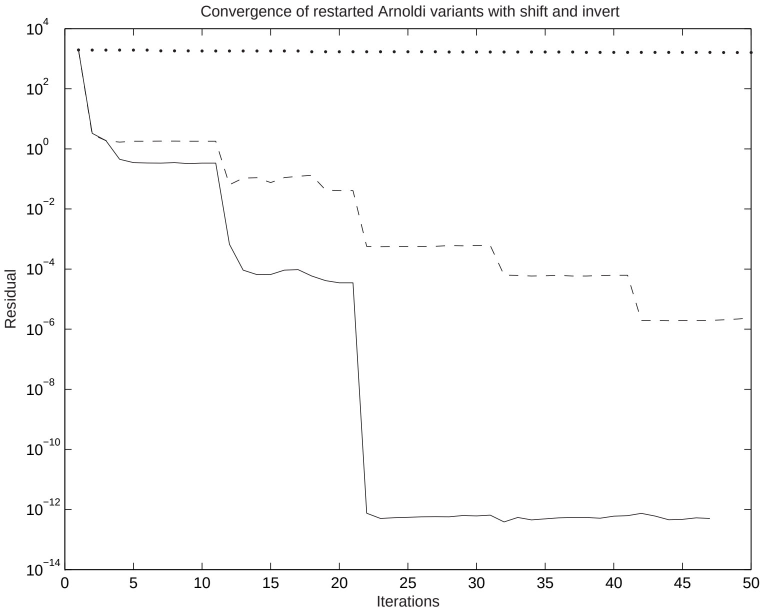


Figure 4.5: Comparison of shift and invert restarted Arnoldi variants

terms of the QEP, which are respectively $O(1)$, $O(\mu)$, and $O(\mu^2)$ in norm. Therefore, since μ is large, the best performance is obtained from Arnoldi variants I (solid line), II (dashed line), and III (dotted line), in that order.

Figure 4.6 shows convergence rates of the Arnoldi variants in a restarted method using simple shifts between restarts instead of a spectral inversion. Observe that Arnoldi I now converges more slowly than the others, as expected (since the Arnoldi I variant neglects the dominant term when μ is small). However, all three methods offer poor convergence, even after 300 iterations; since the desired eigenvalue μ is not well-separated from the rest of the spectrum, poor convergence results.

4.1.3 Convergence of residual-maximizing algorithms

Figure 4.7 shows corresponding convergence rates of four residual-maximizing algorithms using a shift and invert after every 10 iterations. The four methods are

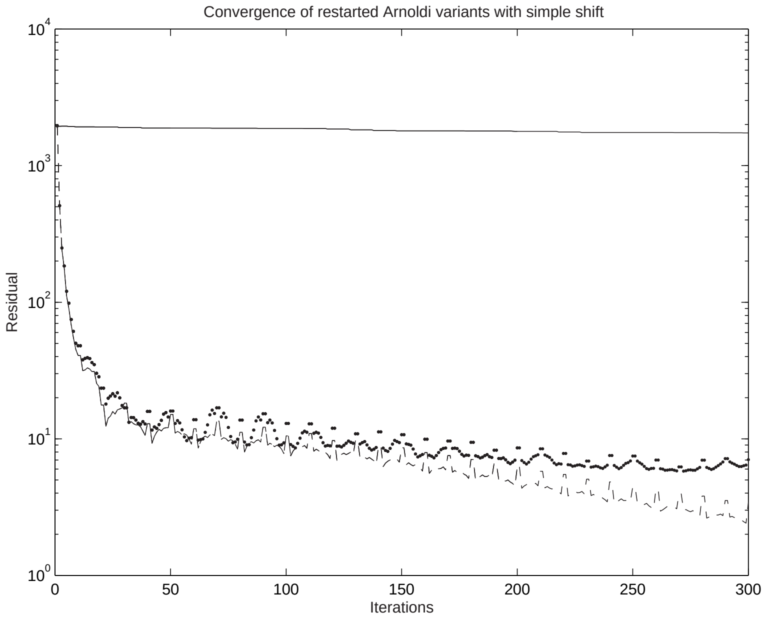


Figure 4.6: Comparison of simply shifted restarted Arnoldi variants

the Arnoldi variant I (solid line), the residual-maximizing algorithm (dashed line) with approximate solution $(\tilde{\mu}_{j+1}, \tilde{u}_{j+1}) = (\mu_j, e_j)$, the Krylov-type projection method from Section 2.2.1 (dotted line), and the residual-maximizing algorithm with natural choice of approximate solution, as in Figure 2.8 (dashed-dotted line). Recall from Theorem 2.5.1 that the Arnoldi I variant and the Krylov-type method are also residual-maximizing algorithms with special choices of approximate solution. Essentially, those algorithms using solutions closest to the natural choice (μ_j, u_j) appear to converge fastest; note that the original residual-maximizing method converges almost completely before the first restart.

Figure 4.8 shows the convergence of the same methods when they are run without restarts, and more clearly illustrates the dramatically different behavior. As in the restarted case, the methods using approximate eigenvector $\tilde{u}_{j+1} = e_j$ (solid and

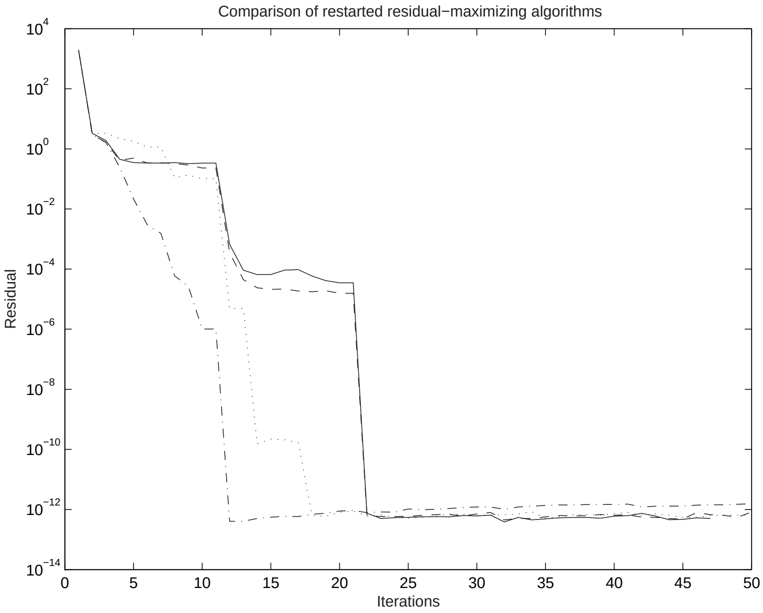


Figure 4.7: Restarted residual-maximizing algorithms

dashed lines) behave comparably; the Krylov-type projection method (with approximate eigenvector $\tilde{u}_{j+1} = e_{\lfloor j/2 \rfloor}$) converges faster, and the natural residual-maximizing method converges fastest.

4.1.4 Behavior of moment-matching projections

To demonstrate the numerical behavior of the Q-Arnoldi, SOAR, and Lanczos-type SOAR methods from Section 3.2, each method is applied to the quadratic acoustics problem of order 564 discussed previously. For this example, we apply a spectral transformation with a real shift, as in Section 4.1.1. Our choice of shift is $\sigma = -260$, so that the symmetric, monic shifted problem in Equation (4.8) can be obtained using a Cholesky factorization. As before, the desired eigenvalue is $\lambda \approx -317.98 + 267.76i$. Figure 4.9 shows the results. Each algorithm is applied for 50 iterations; the gap between the Ritz values and the eigenvalue is plotted in Figure 4.9(a), and

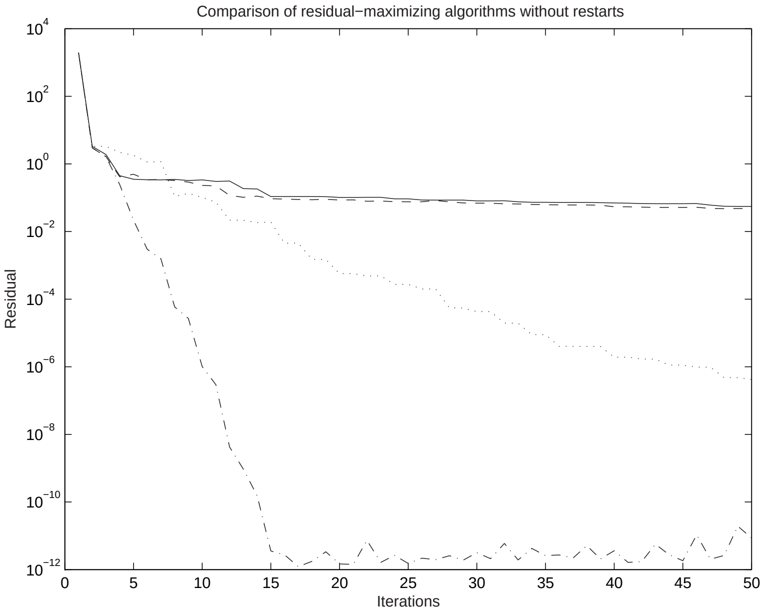


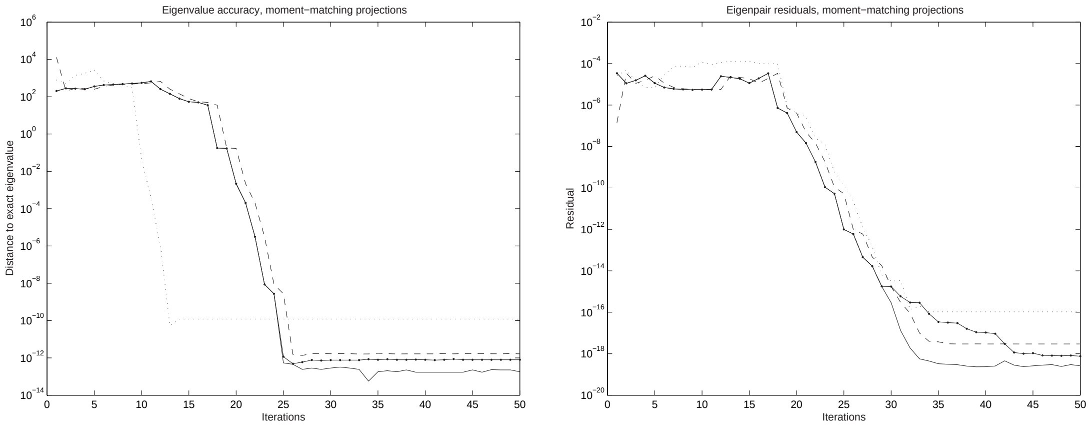
Figure 4.8: Residual-maximizing algorithms, unrestarted

the residuals of the corresponding Ritz pairs appear in Figure 4.9(b). The dashed line shows the residuals computed with the Q-Arnoldi method, while the solid and dotted-solid lines were computed with the SOAR and generalized Lanczos SOAR methods, respectively. All three of these projection methods offer comparable rates of convergence, which is to be expected since they are Arnoldi-based methods matching m moments.

Furthermore, convergence of nonsymmetric Lanczos is shown in the dotted lines of Figure 4.9. Recall from Section 3.3 that a triangular-Hessenberg quadratic eigenvalue problem of order m can be constructed from $2m$ iterations of nonsymmetric Lanczos. The recurrence constructs biorthogonal P_{2m} and Q_{2m} so that

$$P_{2m}^{T}\begin{pmatrix} 0 & I \\ \hat{C} & \hat{B} \end{pmatrix} Q_{2m} = T_{2m}.$$

If (μ, u) is an eigenpair of T_{2m}, then the Ritz pair of $\begin{pmatrix} 0 & I \\ \hat{C} & \hat{B} \end{pmatrix}$ could be defined



(a) Accuracy of Ritz values

(b) Eigenpair residuals

Figure 4.9: Moment-matching projections

reasonably as $(\mu, Q_{2m}u)$, since its residual is orthogonal to P_{2m}. If the Ritz vector $Q_{2m}u$ were an exact eigenvector of the linearization, then it would of course have the form $\begin{pmatrix} x \\ \lambda x \end{pmatrix}$ for an eigenvector x of the original quadratic eigenvalue problem. Therefore, we can choose the last n components $(Q_{2m}u)_{1:n}$ as an approximate eigenvector of the QEP. Similarly, if the triangular-Hessenberg reduction is applied to T_{2m} to construct the linearization $\begin{pmatrix} 0 & I \\ C_m & B_m \end{pmatrix}$, then the eigenpair (μ, v) of this order m quadratic eigenvalue problem gives rise to a corresponding eigenvector $PW^{-1}\begin{pmatrix} v \\ \mu v \end{pmatrix}$ of T_{2m}; the corresponding approximate eigenvector of the original QEP would be $\left(Q_{2m}P\begin{pmatrix} v \\ L^{-1}(\mu I - D_1)v \end{pmatrix}\right)_{1:n}$. According to Figure 4.9, this choice of approximate eigenpair produces residuals that converge like the three Arnoldi-based methods. However, the accuracy of the Ritz values improves much faster; this is expected, since the method matches $4m$ moments instead of m.

Bibliography

[1] P. R. Amestoy, T. A. Davis, and I. S. Duff. An approximate minimum degree ordering algorithm. *SIAM J. Matrix Anal. Appl.*, 17(4):886–905, 1996.

[2] T. Arbogast, M. F. Wheeler, and I. Yotov. Mixed finite elements for elliptic problems with tensor coefficients as cell-centered finite differences. *SIAM J. Numer. Anal.*, 34(2):828–852, 1997.

[3] J. Baglama, D. Calvetti, and L. Reichel. Iterative methods for the computation of a few eigenvalues of a large symmetric matrix. *BIT*, 36(3):400–421, 1996. International Linear Algebra Year (Toulouse, 1995).

[4] J. Baglama, D. Calvetti, and L. Reichel. Algorithm 827: irbleigs: a MATLAB program for computing a few eigenpairs of a large sparse Hermitian matrix. *ACM Trans. Math. Software*, 29(3):337–348, 2003.

[5] Z. Bai, D. Day, and Q. Ye. ABLE: an adaptive block Lanczos method for non-Hermitian eigenvalue problems. *SIAM J. Matrix Anal. Appl.*, 20(4):1060–1082 (electronic), 1999. Sparse and structured matrices and their applications (Coeur d'Alene, ID, 1996).

[6] Z. Bai and Y. Su. A second-order Krylov subspace and its application to the quadratic eigenvalue problem. Technical report, Department of Computer Science, University of California at Davis, September 2003.

[7] Z. Bai and Y. Su. A second-order Krylov subspace and its applications. In *Mini-Workshop: Dimensional Reduction of Large-Scale Systems*. Mathematisches Forschungsinstitut Oberwolfach, October 2003.

[8] A. Bermúdez, R. Durán, M. A. Muschietti, R. Rodríguez, and J. Solomin. Finite element vibration analysis of fluid-solid systems without spurious modes. *SIAM J. Numer. Anal.*, 32(4):1280–1295, 1995.

[9] A. Bermúdez, R. G. Durán, R. Rodríguez, and J. Solomin. Finite element analysis of a quadratic eigenvalue problem arising in dissipative acoustics. *SIAM J. Numer. Anal.*, 38(1):267–291 (electronic), 2000.

[10] D. A. Bini, L. Gemignani, and F. Tisseur. The Ehrlich-Aberth method for the nonsymmetric tridiagonal eigenvalue problem. Numerical Analysis Report 428, Manchester Centre for Computational Mathematics, June 2003.

[11] J. K. Cullum and R. A. Willoughby. *Lánczos algorithms for large symmetric eigenvalue computations. Vol. I: Theory*, volume 3 of *Progress in Scientific Computing*. Birkhäuser Boston Inc., Boston, MA, 1985.

[12] J. K. Cullum and R. A. Willoughby. *Lánczos algorithms for large symmetric eigenvalue computations. Vol. II: Programs*, volume 3 of *Progress in Scientific Computing*. Birkhäuser Boston Inc., Boston, MA, 1985.

[13] D. Day. An efficient implementation of the nonsymmetric Lanczos algorithm. *SIAM J. Matrix Anal. Appl.*, 18(3):566–589, 1997.

[14] C. de Villemagne and R. E. Skelton. Model reductions using a projection formulation. *Internat. J. Control*, 46(6):2141–2169, 1987.

[15] J. Demmel, I. Dhillon, and H. Ren. On the correctness of parallel bisection in floating point. *ETNA*, 3:116–149, 1995.

[16] J. W. Demmel. *Applied numerical linear algebra.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997.

[17] I. S. Dhillon and A. N. Malyshev. Inner deflation for symmetric tridiagonal matrices. *Linear Algebra Appl.*, 358:139–144, 2003. Special issue on accurate solution of eigenvalue problems (Hagen, 2000).

[18] L. C. Evans. *Partial differential equations*, volume 19 of *Graduate Studies in Mathematics.* American Mathematical Society, Providence, RI, 1998.

[19] R. W. Freund and P. Feldmann. The SyMPVL algorithm and its applications to interconnect simulation. Numerical Analysis Manuscript 97-3-04, Bell Laboratories, Murray Hill, New Jersey, June 1997.

[20] S. D. Garvey, F. Tisseur, M. I. Friswell, J. E. T. Penny, and U. Prells. Simultaneous tridiagonalization of two symmetric matrices. *Internat. J. Numer. Methods Engrg.*, 57(12):1643–1660, 2003.

[21] I. Gohberg, P. Lancaster, and L. Rodman. *Matrix polynomials.* Computer Science and Applied Mathematics. Academic Press, New York, 1982.

[22] G. H. Golub and C. F. Van Loan. *Matrix computations.* Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, third edition, 1996.

[23] E. J. Grimme. *Krylov Projection Methods For Model Reduction.* PhD thesis, University of Illinois at Urbana-Champaign, 1997.

[24] N. J. Higham. *Accuracy and stability of numerical algorithms.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1996.

[25] N. J. Higham and F. Tisseur. Bounds for eigenvalues of matrix polynomials. *Linear Algebra Appl.*, 358:5–22, 2003. Special issue on accurate solution of eigenvalue problems (Hagen, 2000).

[26] L. Hoffnung, R.-C. Li, and Q. Ye. Krylov type subspace methods for matrix polynomials. Submitted to Linear Algebra and its Applications, to appear.

[27] L. Hoffnung, R.-C. Li, and Q. Ye. Krylov type subspace methods for matrix polynomials. Technical Report 2002-08, Department of Mathematics, University of Kentucky, 2002.

[28] U. B. Holz. *Subspace approximation methods for perturbed quadratic eigenvalue problems.* PhD thesis, Stanford University, May 2002.

[29] T.-M. Hwang, W.-W. Lin, and V. Mehrmann. Numerical solution of quadratic eigenvalue problems with structure-preserving methods. *SIAM J. Sci. Comput.*, 24(4):1283–1302 (electronic), 2003.

[30] A. V. Knyazev. Toward the optimal preconditioned eigensolver: locally optimal block preconditioned conjugate gradient method. *SIAM J. Sci. Comput.*, 23(2):517–541 (electronic), 2001. Copper Mountain Conference (2000).

[31] T. R. Kowalski. *Extracting a few eigenpairs of symmetric indefinite matrix pencils.* PhD thesis, University of Kentucky, Lexington, KY, 2000.

[32] R. B. Lehoucq, D. C. Sorensen, and C. Yang. *ARPACK users' guide.* Software, Environments, and Tools. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1998. Solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods.

[33] R.-C. Li. Structural preserving model reductions. Research report 2004-02, Department of Mathematics, University of Kentucky, January 2004.

[34] R.-C. Li and Q. Ye. A Krylov subspace method for quadratic matrix polynomials with application to constrained least squares problems. *SIAM J. Matrix Anal. Appl.*, 25(2):405–428, 2003.

[35] K. Meerbergen and M. Robbé. The Arnoldi method for the solution of the quadratic eigenvalue problem and parametrized equations.

[36] V. Mehrmann and D. Watkins. Structure-preserving methods for computing eigenpairs of large sparse skew-Hamiltonian/Hamiltonian pencils. *SIAM J. Sci. Comput.*, 22(6):1905–1925 (electronic), 2000.

[37] V. Mehrmann and D. Watkins. Polynomial eigenvalue problems with Hamiltonian structure. *Electron. Trans. Numer. Anal.*, 13:106–118 (electronic), 2002.

[38] R. B. Morgan. On restarting the Arnoldi method for large nonsymmetric eigenvalue problems. *Math. Comp.*, 65(215):1213–1230, 1996.

[39] K. W. Morton and D. F. Mayers. *Numerical solution of partial differential equations*. Cambridge University Press, Cambridge, 1994.

[40] A. Odabasioglu, M. Celik, and L. T. Pileggi. PRIMA: passive reduced-order interconnect macromodeling algorithm. *IEEE Trans. on CAD*, 17(8), August 1998.

[41] B. N. Parlett. *The symmetric eigenvalue problem*, volume 20 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1998. Corrected reprint of the 1980 original.

[42] B. N. Parlett and H. C. Chen. Use of indefinite pencils for computing damped natural modes. *Linear Algebra Appl.*, 140:53–88, 1990.

[43] B. N. Parlett and I. S. Dhillon. Relatively robust representations of symmetric tridiagonals. In *Proceedings of the International Workshop on Accurate Solution of Eigenvalue Problems (University Park, PA, 1998)*, volume 309, pages 121–151, 2000.

[44] B. N. Parlett and Y. Saad. Complex shift and invert strategies for real matrices. *Linear Algebra Appl.*, 88/89:575–595, 1987.

[45] B. N. Parlett, D. R. Taylor, and Z. A. Liu. A look-ahead Lánczos algorithm for unsymmetric matrices. *Math. Comp.*, 44(169):105–124, 1985.

[46] P.-A. Raviart and J. M. Thomas. A mixed finite element method for 2nd order elliptic problems. In *Mathematical aspects of finite element methods (Proc. Conf., Consiglio Naz. delle Ricerche (C.N.R.), Rome, 1975)*, pages 292–315. Lecture Notes in Math., Vol. 606. Springer, Berlin, 1977.

[47] R. Rodríguez and J. E. Solomin. The order of convergence of eigenfrequencies in finite element approximations of fluid-structure interaction problems. *Math. Comp.*, 65(216):1463–1475, 1996.

[48] Y. Saad. Projection methods for solving large sparse eigenvalue problems. In B. Kågström and A. Ruhe, editors, *Matrix Pencils Proceedings, Pite Havsbad, 1982*, number 973 in Lecture Notes in Mathematics, pages 121–144, New York, 1983. Springer-Verlag.

[49] Y. Saad. *Numerical methods for large eigenvalue problems.* Algorithms and Architectures for Advanced Scientific Computing. Manchester University Press, Manchester, 1992.

[50] B. T. Smith, J. M. Boyle, J. J. Dongarra, B. S. Garbow, Y. Ikebe, V. C. Klema, and C. B. Moler. *Matrix Eigensystem Routines - EISPACK Guide.* Springer-Verlag, 1976.

[51] S. I. Solov′ev. Preconditioned iterative methods for monotone nonlinear eigenvalue problems. Preprint-Reihe des Chemnitzer SFB393/03-08, Technische Universität Chemnitz, Chemnitz, Germany, March 2003.

[52] M. Stewart and G. W. Stewart. On hyperbolic triangularization: stability and pivoting. *SIAM J. Matrix Anal. Appl.*, 19(4):847–860 (electronic), 1998.

[53] T.-J. Su and R. R. Craig Jr. Model reduction and control of flexible structures using Krylov vectors. *J. Guidance Control Dynamics*, 14(2):260–267, 1991.

[54] F. Tisseur. Tridiagonal-diagonal reduction of symmetric indefinite pairs. Numerical Analysis Report 409, Manchester Centre for Computational Mathematics, November 2003. To appear in SIAM J. Matrix Anal. Appl.

[55] F. Tisseur and K. Meerbergen. The quadratic eigenvalue problem. *SIAM Rev.*, 43(2):235–286 (electronic), 2001.

[56] Q. Ye. A convergence analysis for nonsymmetric Lanczos algorithms. *Math. Comp.*, 56(194):677–691, 1991.

[57] Q. Ye. A breakdown-free variation of the nonsymmetric Lanczos algorithms. *Math. Comp.*, 62(205):179–207, 1994.

[58] Q. Ye. An iterated shift-and-invert Arnoldi algorithm for quadratic matrix eigenvalue problems. In *Symposium on Scientific Computing*. University of Hong Kong, 2004.

Leonard Hoffnung

Vita

Born September 5, 1974 in Chicago, Illinois

Education

- M.S., Mathematics, University of Kentucky, Lexington, KY. May 1999.
- B.A. *summa cum laude*, Mathematics, Southern Illinois University, Carbondale, IL. May 1997.

Experience

- Graduate research assistant, Los Alamos National Laboratory, Los Alamos, NM. Summer 2002 and Summer 2003.
- Teaching assistant, University of Kentucky. 1998–2004.
- Linux/Unix system administrator, Math Sciences Computing Facility, University of Kentucky. Fall 2000 - Spring 2001.

Publications

- L. Hoffnung, R.-C. Li, Q. Ye, *"Krylov type subspace methods for matrix polynomials."* (To appear in Linear Algebra and its Applications.)
- A. Beltukov, J. Choi, L. Hoffnung, N. Nigam, D. Sterling, P. Tupper, *"Problems in ultra-high-precision GPS position estimation."* (IMA Preprint Series, 1998).

Awards

- Full Tuition Presidential Fellowship, University of Kentucky. 2001-2002.
- Multi-year Quality Achievement Fellowship, University of Kentucky. 1997-2000.
- Full Tuition Open Competition Fellowship, University of Kentucky. 1997-1998. Awarded to incoming graduate students who show exceptional promise.
- Townsend Award, Southern Illinois University, 1996 and 1997. Awarded each year to the top junior and the top senior in mathematics.
- Full Tuition Presidential Scholarship, Southern Illinois University, 1993-1997. Awarded to outstanding incoming freshmen.