Computations and Analysis with Non-normal Matrices

Natalie Wellen

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2025

Reading Committee:

Anne Greenbaum, Chair

J. Nathan Kutz

Heather Wilber

Program Authorized to Offer Degree:
Applied Mathematics

University of Washington

Abstract

Computations and Analysis with Non-normal Matrices

Natalie Wellen

Chair of the Supervisory Committee:
Anne Greenbaum
Department of Applied Mathematics

The goal of this dissertation is to introduce some new iterative methods for solving problems involving non-Hermitian matrices and to add to our understanding of the convergence of these algorithms when applied to highly non-normal matrices. In the first part, the Arnoldi-OR algorithm [23] is introduced. Given an n by n matrix $\mathbf{A}$ and a rational function $N(z)/D(z)$, where $D(\mathbf{A})$ is nonsingular, this algorithm finds the approximations x_k, $k = 1, 2, \ldots$, from successive *Krylov subspaces*, $\text{span}\{\mathbf{b}, \mathbf{Ab}, \ldots, \mathbf{A}^{k-1}\mathbf{b}\}$, that minimize the 2-norm of the residual, $\|D(\mathbf{A})\mathbf{x}_k - N(\mathbf{A})\mathbf{b}\|_2$. Convergence of the Arnoldi-OR algorithm can be bounded based on the eigenvalues of $\mathbf{A}$ and the condition number of the best-conditioned matrix of eigenvectors, assuming that $\mathbf{A}$ is diagonalizable. This may be a large overestimate, however, if the best-conditioned matrix of eigenvectors is still very ill-conditioned.

Starting in Chapter 4, the second part of this dissertation explores bounds on the norm of a function of $\mathbf{A}$ that can be applied when $\mathbf{A}$ is *highly non-normal*; i.e., either $\mathbf{A}$ is not diagonalizable or it is diagonalizable but the eigenvalues are very ill-conditioned. These bounds involve the ∞-norm of the function, not just on the eigenvalues of $\mathbf{A}$, but on a larger set in the complex plane containing the eigenvalues. In the second part of the dissertation, we expand on some known results [30, 29] about K-spectral sets—sets $\Omega \subset \mathbb{C}$ satisfying $\|f(\mathbf{A})\|_2 \leq K \sup_{z\in\Omega} |f(z)|$ for all functions f analytic in Ω, (i.e. functions that can be arbitrarily well-approximated on Ω by rational functions with no poles in Ω). This work is described in [61]. Here, we use it to give alternative bounds on the 2-norm of the residual

in the Arnoldi-OR algorithm.

We also consider a different way of solving non-Hermitian linear systems, which is to convert the problem to a Hermitian one, in Section 2.2. If one can find Hermitian positive definite matrices $\mathbf{M}$ and $\mathbf{Y}$ such that $\mathbf{A} = \mathbf{M}^{-1}\mathbf{Y}$, then one can solve $\mathbf{Ax} = \mathbf{b}$ using the conjugate gradient method (CG) applied to the system $\mathbf{Yx} = \mathbf{Mb}$ with $\mathbf{M}$ as a preconditioner. We apply this technique to a problem involving the graph Laplacian that is of interest to Sandia National Laboratory, where I worked with Richard Lehoucq over the summer of 2023.

TABLE OF CONTENTS

NOTATION

Notation	Definition
α, a	Scalar values are represented by lower case Greek or Roman letters and are assumed to be complex unless otherwise stated, $\alpha, a \in \mathbb{C}$.
$\mathbf{b}$	Vectors are denoted by bold lower case letters and are assumed to be n-dimensional with complex entries unless otherwise stated, $\mathbf{b} \in \mathbb{C}^n$.
$\|\mathbf{x}\|_2$	The 2-norm of $\mathbf{x}$: $\|\mathbf{x}\|_2 = \left(\sum_{j=1}^{n} \|x_j\|^2\right)^{1/2}$.
$\langle \mathbf{y}, \mathbf{x} \rangle$	The Euclidean inner product of $\mathbf{y}$ and $\mathbf{x}$: $\langle \mathbf{y}, \mathbf{x} \rangle = \mathbf{y}^* \mathbf{x} = \sum_{j=1}^{n} \bar{y}_j x_j$.
$\|\mathbf{x}\|_{\mathbf{A}}$	The $\mathbf{A}$-norm of $\mathbf{x}$, where $\mathbf{A}$ is Hermitian positive definite. $\|\mathbf{x}\|_{\mathbf{A}} = \langle \mathbf{x}, \mathbf{A}\mathbf{x} \rangle^{1/2}$.
$\mathbf{A}$	Matrices are denoted by bold capital letters.
$\mathbf{A}^T$	The transpose of $\mathbf{A}$: $\mathbf{A}^T(i,j) = \mathbf{A}(j,i)$
$\mathbf{A}^*$	The Hermitian transpose of $\mathbf{A}$: $\mathbf{A}^*(i,j) = \overline{\mathbf{A}(j,i)}$.
$\kappa_k(\mathbf{A}, \mathbf{b})$	The k-dimensional Krylov subspace $\text{span}\{\mathbf{b}, \mathbf{A}\mathbf{b}, \ldots, \mathbf{A}^{k-1}\mathbf{b}\}$.
$\|\mathbf{A}\|_2$	The matrix norm corresponding to the 2-norm for vectors: $\|\mathbf{A}\|_2 = \max_{\|\mathbf{v}\|_2 = 1} \|\mathbf{A}\mathbf{v}\|_2$.
$\kappa(\mathbf{A})$	The 2-norm condition number of $\mathbf{A}$: If $\mathbf{A}$ is invertible, then $\kappa(\mathbf{A}) = \|\mathbf{A}\|_2 \, \|\mathbf{A}^{-1}\|_2$.
$W(\mathbf{A})$	The numerical range of $\mathbf{A}$: $W(\mathbf{A}) = \{\langle \mathbf{q}, \mathbf{A}\mathbf{q} \rangle : \langle \mathbf{q}, \mathbf{q} \rangle = 1\}$.
$w(\mathbf{A})$	The numerical radius of $\mathbf{A}$: $w(\mathbf{A}) = \max_{z \in W(\mathbf{A})} \|z\|$.
$\rho(\mathbf{A})$	The numerical abscissa of $\mathbf{A}$: $\rho(\mathbf{A}) = \max_{z \in W(\mathbf{A})} \text{Re}(z)$.
$\mathcal{D}(c, r)$	The open disk centered at c of radius r.
$\partial\Omega$	If Ω is a region in the complex plane, then $\partial\Omega$ is its boundary.

DEDICATION

to my parents, John and Marcia Wellen

and my uncle, Stephen A Stohlman

Chapter 1

INTRODUCTION

The goal of this dissertation is to introduce some new iterative methods for solving problems involving non-Hermitian matrices, and to add to our understanding of the convergence of these algorithms when applied to highly non-normal matrices. In the first part of this dissertation, I introduce a new method for solving a non-Hermitian system of linear equations with a three-term recurrence method when that system satisfies certain conditions. I also introduce an iterative method for solving the action of a rational function of a matrix on a vector. This iterative method is a Krylov subspace method based on the Krylov subspace generated by $\mathbf{A}$ and $\mathbf{b}$, and has a priori bounds on the convergence of its error. In the second part of this dissertation, I explore bounds on the norm of a function of $\mathbf{A}$ that can be applied when $\mathbf{A}$ is *highly non-normal*, such as when the eigenvalues are highly ill-conditioned. I expand on known results and compare different methods for calculating an upper bound on the norm of a function of a matrix.

Chapter 2 gives background material on Krylov subspace methods, including the conjugate gradient method (CG) [70], the generalized minimal residual algorithm (GMRES) [96], and the biconjugate gradient stabilized method (Bi-CGSTAB) [112]. The relationship between CG and the Lanczos algorithm and that between GMRES and the Arnoldi algorithm is summarized. Here, we also review preconditioning and how a preconditioner may be used to transform a non-Hermitian problem into an equivalent Hermitian one. In particular, we discuss the solution of a linear system involving the graph Laplacian.

Chapter 3 explores the problem of computing the product of a rational function $R(\mathbf{A}) = D(\mathbf{A})^{-1}N(\mathbf{A})$ and a vector $\mathbf{b}$ [23]. Here, approximate solutions are chosen from a Krylov space based on $\mathbf{A}$ and $\mathbf{b}$, $\text{span}\{\mathbf{b}, \mathbf{A}\mathbf{b}, \ldots, \mathbf{A}^{k-1}\mathbf{b}\}$, instead of one based on $D(\mathbf{A})$, which is

used when GMRES is applied to $D(\mathbf{A})\mathbf{x} = N(\mathbf{A})\mathbf{b}$. The approximate solution $\mathbf{x}_k$ at step k is chosen to minimize the 2-norm of the residual, $\|D(\mathbf{A})\mathbf{x}_k - N(\mathbf{A})\mathbf{b}\|_2$. This method is closely related to the Arnoldi-FA algorithm [21], but the Arnoldi-FA algorithm does not minimize any standard error norm. Thus, the 2-norm of the residual at each step in Arnoldi-OR is guaranteed to be less than or equal to that in Arnoldi-FA. Numerical experiments suggest, however, that the 2-norm of the error, $\|\mathbf{x}_k - R(\mathbf{A})\mathbf{b}\|_2$, in the two methods is very similar. We do not yet have a complete explanation for this similarity, and it is listed in Chapter 6 as a possibile area for future work. Computing $R(\mathbf{A})\mathbf{b}$ for general rational functions, $R(z)$, and non-Hermitian matrices, $\mathbf{A}$, is relevant to fields that use exponential integrators to solve ordinary differential equations of the form $\frac{d}{dt}u(t) = F(u(t)),\ u(t_0) = u_0$ [46], the matrix sine and cosine functions [66], the matrix exponential [82], and more. Note that no matrix-matrix multiplication is required since $D(\mathbf{A})$ and $N(\mathbf{A})$ can be applied to vectors without actually forming the matrices $D(\mathbf{A})$ and $N(\mathbf{A})$. The MATLAB codes for these algorithms can be found at `https://www.github.com/tygris/ArnoldiOR`.

Chapter 4 gives background material on non-normal matrices and bounding $\|f(\mathbf{A})\|_2$, including when $f(\mathbf{A})$ is a rational function like $R(\mathbf{A})$. For normal matrices, eigenvalues will determine this bound. Eigenvalues, a powerful tool from linear algebra, generally play a significant role in understanding and solving complex problems. For example, the QR algorithm to compute eigenvalues is listed as one of the top ten algorithms of the 20th century [26]. Eigenvalues are calculated to decompose systems in a way that makes it easier to solve a problem, study resonance in physical systems, and analyze a system's asymptotic behavior. A *normal matrix* has a complete set of orthogonal eigenvectors. In the study of resonance this property leads to well-understood system perturbations, for instance, and means that a system's short-term behavior matches its asymptotic behavior. When a matrix is just slightly non-normal, meaning that it has a complete set of eigenvectors that are fairly close to orthogonal, then eigenvalues still play a vital role. However, when a matrix is highly non-normal, meaning that either it is not diagonalizable or its eigenvectors are almost linearly dependent, eigenvalues may be misleading. In the first chapter of *Spectra*

and Pseudospectra [108], Trefethen and Embree list 19 examples of scientific fields that rely on eigenvalue results. Then, in the second chapter of *Spectra and Pseudospectra*, Trefethen and Embree list:

atmospheric sciences [45]	control theory [71]
ecology [83]	hydrodynamic stability [110]
lasers [97]	magnetohydrodynamics [11]
Markov chains [76]	matrix iterations [59]
rounding error analysis [20]	operator theory [15]
non-Hermitian quantum mechanics [67]	
numerical solution of differential equations [113]	

as examples of misleading eigenvalue analysis from different scientific fields. This dissertation addresses some of the difficulties that arise from numerical computations involving highly non-normal matrices.

The second part of this dissertation is about K-spectral sets and how they can be used to bound the norms of functions of highly non-normal matrices. Let Ω be any region of the complex plane containing the eigenvalues of $\mathbf{A}$. The region Ω may be convex, the union of many disconnected subsets, a half-plane, or any other shape. Ω is said to be a K-spectral set for $\mathbf{A}$ if $\|f(\mathbf{A})\|_2 \leq K \sup_{z\in\Omega} |f(z)|$ for all functions f analytic in Ω. If $K = 1$, then Ω is referred to simply as a *spectral set* for $\mathbf{A}$.

Although this dissertation deals with finite matrices, research about K-spectral sets originated in the mathematical field of functional analysis. In 1950, Von Neumann applied spectral set theory to contraction linear operators, and clearly connected spectral set theory to matrices (aka finite dimensional linear operators) [116]. There have also been more recent advancements in K-spectral set research. In 2007, Crouzeix published a paper showing that the numerical range of a linear operator is a K-spectral set for a value of K bounded by 11.08 [28]. For a matrix $\mathbf{A}$, the numerical range is defined as

$$W(\mathbf{A}) = \left\{ \frac{\mathbf{x}^*\mathbf{A}\mathbf{x}}{\mathbf{x}^*\mathbf{x}} | \mathbf{x} \in \mathbb{C}^n, \mathbf{x} \neq \mathbf{0} \right\}.$$

A similar definition holds for other linear operators, using inner products in an appropriate Hilbert space. It is well-known that the numerical range of a matrix $\mathbf{A}$ is always a convex set containing the eigenvalues of $\mathbf{A}$ [74], and Crouzeix's result shows that $W(\mathbf{A})$ is an 11.08-spectral set for $\mathbf{A}$. While 11.08 is a bound that only allows the norm of the function of a matrix to exceed by one order of magnitude the ∞-norm of the function on the numerial range, Crouzeix conjectured that $K = 2$ is the optimal value of K [28]. This drew more researchers to study K-spectral sets.

As of 2025, $K = 2$ for the numerical range of all linear operators remains unproven. However, for specific classes of matrices, the numerical range has been proven to be a 2-spectral set. These classes of matrices include 2×2 matrices [27], 3×3 tridiagonal matrices with a single value along the diagonal [52], and matrices of the form $\alpha\mathbf{I} + \mathbf{DP}$ where $\mathbf{D}$ is a diagonal matrix and $\mathbf{P}$ is a permutation matrix [25].

For the general case of all linear operators, in 2017 Crouzeix and Palencia proved that the numerical range is a $1 + \sqrt{2}$, or about 2.414, spectral set [30]. Then, in 2019 Crouzeix and Greenbaum showed that one can remove a single disk from the numerical range and if the radius of the disk is less than a certain value then the remaining set is a $(3 + 2\sqrt{3})$-spectral set [29]. They used this K-spectral set to give bounds on the 2-norm of the residual in the GMRES algorithm when the numerical range of $\mathbf{A}$ contains the origin. Chapter 5 generalizes this result to cover the removal of any number of disks, and we present several examples where this might prove useful. The analytically calculated value of K is based on the number of disks removed when the radius of the removed disk satisfies one of two bounds. Another approach to obtain bounds on $\|f(\mathbf{A})\|$ is to use the Cauchy integral formula,

$$f(\mathbf{A}) = \frac{1}{2\pi i} \oint_{\partial\Omega} (z\mathbf{I} - \mathbf{A})^{-1} f(z)\, dz, \tag{1.1}$$

and replace the norm of the integral by the integral of the norm of the integrand:

$$\|f(\mathbf{A})\| \leq \frac{1}{2\pi} \left(\oint_{\partial\Omega} \|(z\mathbf{I} - \mathbf{A})^{-1}\|\, |dz| \right) \sup_{z\in\partial\Omega} |f(z)|, \tag{1.2}$$

also known as the integral of the resolvent norm.

Chapter 5 contains extensions to the basic theorem from [29] that can be applied to an arbitrary region Ω, whose interior contains the eigenvalues of $\mathbf{A}$. When $f(z)$ is analytic in Ω, then we explain how to use those extensions to obtain a bound on $\|f(\mathbf{A})\|$ that *may be* smaller than that in (1.2). These extensions describe how to calculate bounds of this form for more types of regions Ω, including the left half-plane, the unit disk, and the 2-norm ϵ-*pseudospectrum* of A; that is, $\Omega = \{z \in \mathbb{C} : \|(z\mathbf{I} - \mathbf{A})^{-1}\|_2 > \epsilon^{-1}\}$. Aside from some minor factors involving the geometry of the region, the extensions to the basic theorem from [29] can be used to numerically determine a bound similar to (1.2), but with the norm of the resolvent, $\|(z\mathbf{I} - \mathbf{A})^{-1}\|$, replaced by twice the absolute value of a certain point on the boundary of the numerical range of the resolvent. We compare these two methods of bounding K for a variety of sets, and also partially explain the mathematical relationship between these upper bounds on K for K-spectral sets. The MATLAB codes used to compute these bounds can be found at `https://github.com/tygris/k-spectral-sets`.

Finally, Chapter 6 contains open research problems related to the work throughout this dissertation.

For the rest of this dissertation, we assume that $\mathbf{A} \in \mathbb{C}^{n\times n}$, so that $\mathbf{A}$ is always a square matrix and we are always working in a complex vector field, unless specifically stated otherwise. Further, we assume that all analysis and discussion of algorithms are in exact arithmetic, unless explicitly stated otherwise.

Chapter 2

KRYLOV SUBSPACE METHODS

Krylov subspace methods are iterative algorithms used to solve linear systems of equations, $\mathbf{Ax} = \mathbf{b}$. These methods are an alternative to direct methods like Gaussian elimination. When $\mathbf{A}$ is a non-singular $n \times n$ matrix, Gaussian elimination uses approximately $\frac{2}{3}n^3$ operations to solve $\mathbf{Ax} = \mathbf{b}$. Gaussian elimination has an arithmetic complexity of $\mathcal{O}(n^3)$, because the largest term is a constant times n^3. The cubic growth of operations in Gaussian elimination matches the worst-case performance of Krylov subspace methods for a dense matrix. However, in most cases, Krylov subspace methods are more computationally feasible than Gaussian elimination for large systems.

The Krylov subspace of a matrix $\mathbf{A}$ and the vector $\mathbf{b}$ is the set of all linear combinations of the vectors contained in $\{\mathbf{b}, \mathbf{Ab}, \mathbf{A}^2\mathbf{b}, \cdots\}$, or the span of this set. The *kth*-Krylov subspace is span$\{\mathbf{b}, \mathbf{Ab}, \mathbf{A}^2\mathbf{b}, \cdots, \mathbf{A}^{k-1}\mathbf{b}\}$. By definition, a Krylov subspace is defined by both a matrix and a vector of interest.

The Cayley-Hamilton Theorem states that for all square matrices, there exists a characteristic polynomial of $\mathbf{A}$ equivalent to the zero matrix[1]. By rearranging the characteristic polynomial, we can write the polynomial $p(\mathbf{A}) = \mathbf{A}^{-1}$. Krylov subspace methods take advantage of this fact by iteratively increasing the degree of the polynomial considered. Further, since matrix-vector multiplication is more efficient to compute than matrix-matrix multiplication, these methods avoid the costly route of computing $p(\mathbf{A})$ by instead computing $p(\mathbf{A})\mathbf{b}$.

Many Krylov subspace methods can be derived from or contain the Lanczos and Arnoldi algorithms as subroutines. The Lanzos algorithm, one version of which is found in Al-

[1]This is the same characteristic polynomial that defines the eigenvalues of a matrix.

Algorithm 1 Lanczos Algorithm

Inputs: $\mathbf{A} \in \mathbb{C}^{n\times n}$ where $\mathbf{A}^* = \mathbf{A}$, initial vector $\mathbf{b} \in \mathbb{C}^n$, and the number of iterations $m \leq n$.
Set $\mathbf{q}_1 = \mathbf{b}/\|\mathbf{b}\|_2$
$\beta_0 := 0,\ \mathbf{q}_0 := \mathbf{0}$
for $k = 1, 2, \cdots, m$ **do**
    $\tilde{\mathbf{q}}_{k+1} = \mathbf{A}\mathbf{q}_k - \beta_{k-1}\mathbf{q}_{k-1}$
    $\alpha_k = \langle \tilde{\mathbf{q}}_{k+1}, \mathbf{q}_k \rangle$
    $\tilde{\mathbf{q}}_{k+1} \leftarrow \tilde{\mathbf{q}}_{k+1} - \alpha_k \mathbf{q}_k$
    $\beta_k = \langle \tilde{\mathbf{q}}_{k+1}, \tilde{\mathbf{q}}_{k+1} \rangle^{1/2}$
    $\mathbf{q}_{k+1} = \tilde{\mathbf{q}}_{k+1}/\beta_k$
end for
Return the matrix of orthonormal basis vectors $\mathbf{Q}_{m+1} = [\mathbf{q}_1\ \mathbf{q}_2\ \cdots \mathbf{q}_{m+1}]$ and the coefficient $\mathbf{T}_{m+1,m}$ where $\mathbf{T}_{m+1,m}$ is a tridiagonal matrix with $\{\alpha_k\}_{k\leq m}$ on the main diagonal, $\{\beta_k\}_{k\leq m-1}$ on the super- and sub- diagonals, and $\mathbf{T}(m+1, m) = \beta_m$.

Algorithm 2 Arnoldi Algorithm

Inputs: $\mathbf{A} \in \mathbb{C}^{n\times n}$, initial vector $\mathbf{b} \in \mathbb{C}^n$, and the number of iterations m.
Set $\mathbf{q}_1 = \mathbf{b}/\|\mathbf{b}\|_2$.
for $j = 1, \ldots, m$ **do**
    $\tilde{\mathbf{q}}_{j+1} = \mathbf{A}\mathbf{q}_j$
    for $i = 1, \ldots, j$ **do**
        $\mathbf{H}(i, j) = \langle \mathbf{q}_i, \tilde{\mathbf{q}}_{j+1} \rangle$
        $\tilde{\mathbf{q}}_{j+1} \leftarrow \tilde{\mathbf{q}}_{j+1} - \mathbf{H}(i, j)\mathbf{q}_i$
    end for
    $\mathbf{H}(j+1, j) = \langle \tilde{\mathbf{q}}_{j+1}, \tilde{\mathbf{q}}_{j+1} \rangle^{1/2}$
    $\mathbf{q}_{j+1} = \tilde{\mathbf{q}}_{j+1}/\mathbf{H}(j+1, j)$
end for
Return the matrix of orthonormal basis vectors $\mathbf{Q}_{m+1} = [\mathbf{q}_1\ \mathbf{q}_2\ \cdots \mathbf{q}_{m+1}]$ and $\mathbf{H}_{m+1,m}$ the upper Hessenberg coefficient matrix defined by the elements $\{\mathbf{H}(i, j)\}_{1\leq i\leq m+1, 1\leq j\leq m}$.

gorithm 1, is a method to compute the matrices $\mathbf{Q}_{m+1}$, $\mathbf{Q}_m$, and $\mathbf{T}_{m+1,m}$ such that $\mathbf{A}\mathbf{Q}_m = \mathbf{Q}_{m+1}\mathbf{T}_{m+1\times m}$ when $\mathbf{A}$ is Hermitian. The definition of a symmetric matrix is $\mathbf{A} = \mathbf{A}^T$. The definition of a Hermitian matrix is $\mathbf{A} = \mathbf{A}^*$. All symmetric and Hermitian matrices must be square, and all real-valued Hermitian matrices are symmetric. When Lanczos is run for m iterations, the matrix $\mathbf{Q} \in \mathbb{C}^{n\times m}$ contains the m orthonormal basis vectors for $\mathcal{K}_m(\mathbf{A}, \mathbf{b}) = \text{span}\{\mathbf{b}, \mathbf{A}\mathbf{b}, \cdots, \mathbf{A}^{m-1}\mathbf{b}\}$, and the triadiagonal matrix $\mathbf{T}_{m+1,m} \in \mathbb{R}^{m+1\times m}$ contains the recurrence coefficients from the Lanczos algorithm.

Unlike Lanczos, the Arnoldi algorithm takes any square matrix as an input, including highly non-normal matrices. One version of Arnoldi is found in Algorithm 2. The Arnoldi algorithm run for n iterations decomposes any matrix $\mathbf{A}$ into $\mathbf{A} = \mathbf{Q}\mathbf{H}\mathbf{Q}^*$, where $\mathbf{H}$ is an $n \times n$ upper Hessenberg matrix. When Arnoldi is run for $m < n$ iterations, then $\mathbf{A}\mathbf{Q}_m = \mathbf{Q}_{m+1}\mathbf{H}_{m+1\times m}$. To form an orthonormal basis for $\mathcal{K}_m(\mathbf{A}, \mathbf{b})$ for the non-Hermitian matrix $\mathbf{A}$, Arnoldi orthogonalizes the next basis vector $\mathbf{q}_{m+1}$ against all $\mathbf{q}_j$ for $j \leq m$. This full orthogonalization results in the upper Hessenberg coefficient matrix output by Arnoldi.

Both Lanczos and Arnoldi form an orthonormal basis of $\mathcal{K}(\mathbf{A}, \mathbf{b})$, and theoretically both these algorithms do the same thing. In fact, running Arnoldi on a Hermitian matrix will lead to the exact same outputs as the Lanczos algorithm. To see this result, one may notice that Arnoldi outputs matrices such that $\mathbf{Q}_n^*\mathbf{A}\mathbf{Q}_n = \mathbf{H}_n$. This equation involves a $*$-congruence transformation of $\mathbf{A}$, which maintains the Hermitian (or lack thereof) property of the matrix being transformed. The $*$-congruence transformation implies that when $\mathbf{A}^* = \mathbf{A}$ then $\mathbf{H}_n^* = \mathbf{H}_n$, which is only possible when $\mathbf{H}_n$ is a Hermitian tridiagonal matrix, which we called $\mathbf{T}_n$. A tridiagonal recurrence coefficient matrix implies the next basis vector, $\mathbf{q}_{k+1}$, only needs to be orthogonalized against the two most recent vectors, $\mathbf{q}_k$ and $\mathbf{q}_{k-1}$, to be orthogonal to all $\mathbf{q}_j$ for $j < k - 1$. This property is known as a 3-term recurrence. Lanczos takes advantage of the 3-term recurrence and symmetric recurrence coefficient matrix for Hermitian matrices to ignore the extra computations that one would otherwise perform by running Arnoldi on the same matrix.

2.1 Krylov Subspace Methods for Solving Linear Systems of Equations

Three popular methods for solving systems of linear equations are conjugate gradient (CG), generalized minimal residual (GMRES), and bi-conjugate gradient stabilized (Bi-CGSTAB). There are many Krylov subspace methods for solving $\mathbf{Ax} = \mathbf{b}$, all of which iteratively increase the dimension of the subspace searched for solutions. For example, CG is derived from Lanczos and during each iteration it minimizes the $\mathbf{A}$-norm of the error restricted to the range of $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$. GMRES uses Arnoldi and minimizes the 2-norm of the residual restricted to the range of $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ during the kth iteration. Other Krylov subspace algorithms for non-Hermitian matrices like Bi-CGSTAB reduce the computational costs by no longer optimizing the error or residual in a norm of interest, but still returning an approximate solution $\mathbf{x}_k$ (hopefully) near optimal.

CG was first introduced by Hestenes and Stieffel (1952) [70]. For guaranteed convergence, CG requires a Hermitian positive definite (HPD) matrix as the input. A matrix is a positive definite matrix when, for all vectors $\mathbf{x}$ not equal to the zero vector, $\mathbf{x}^*\mathbf{Ax} > 0$. When the matrix $\mathbf{A}$ is HPD, all the eigenvalues of $\mathbf{A}$ are real and strictly greater than zero. CG can be derived in multiple ways, including from the Lanczos algorithm or from the steepest descent algorithm from optimization; see Greenbaum (1997) [57] or Saad (2003) [95] for more details.

Algorithm 3 Conjugate Gradient Algorithm

Inputs: HPD $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\mathbf{b} \in \mathbb{C}^n$, the tolerance ϵ, and the initial guess $\mathbf{x}_0 \in \mathbb{C}^n$
$\mathbf{r}_0 = \mathbf{b} - \mathbf{Ax}_0$; $\mathbf{p}_0 = \mathbf{r}_0$
$k = 0$
while $\|\mathbf{r}_k\|_2 > \epsilon$ **do**
    $\mu_k \leftarrow \langle \mathbf{r}_k, \mathbf{r}_k \rangle / \langle \mathbf{Ap}_k, \mathbf{p}_k \rangle$
    $\mathbf{x}_{k+1} \leftarrow \mathbf{x}_k + \mu_k \mathbf{p}_k$
    $\mathbf{r}_{k+1} \leftarrow \mathbf{r}_k - \mu_k \mathbf{Ap}_k$
    $\tau_k \leftarrow \langle \mathbf{r}_{k+1}, \mathbf{r}_{k+1} \rangle / \langle \mathbf{r}_k, \mathbf{r}_k \rangle$
    $\mathbf{p}_{k+1} \leftarrow \mathbf{r}_{k+1} + \tau_k \mathbf{p}_k$
    $k \leftarrow k + 1$
end while
Return $\mathbf{x}_k$

At the core of Algorithm 1 and Algorithm 3 (the conjugate gradient algorithm on the next page) are 3-term recurrences. A 3-term recurrence in Algorithm 1 can be seen by plugging in $\tilde{\mathbf{q}}_k$ from lines 5 and 7 into line 9, resulting in

$$\beta_k \mathbf{q}_{k+1} = \mathbf{A}\mathbf{q}_k - \alpha_k \mathbf{q}_k - \beta_{k-1}\mathbf{q}_{k-1}.$$

Likewise, from Algorithm 3, we have that

$$\begin{aligned} \mathbf{p}_k &= \mathbf{r}_k + \tau_{k-1}\mathbf{p}_{k-1}, \\ \Rightarrow \mathbf{r}_k &= \mathbf{p}_k - \tau_{k-1}\mathbf{p}_{k-1}, \text{ and }, \\ \mathbf{r}_{k+1} &= \mathbf{r}_k - \mu_k \mathbf{A}\mathbf{p}_k \\ &= -\mu_k \mathbf{A}\mathbf{p}_k + \mathbf{p}_k - \tau_{k-1}\mathbf{p}_{k-1}. \end{aligned}$$

By replacing $\mathbf{r}_{k+1}$ in line 9 with the above relationship, then $\mathbf{p}_{k+1}$ can be rewritten as the three term recurrence

$$\mathbf{p}_{k+1} = -\mu_k \mathbf{A}\mathbf{p}_k + (1+\tau_k)\mathbf{p}_k - \tau_{k-1}\mathbf{p}_{k-1}.$$

Further, $\mathbf{r}_{k+1}$ can also be written as the 3-term recurrence

$$\mathbf{r}_{k+1} = \left(1 - \frac{\mu_k}{\mu_{k-1}}\tau_{k-1}\right)\mathbf{r}_k - \mu_k \mathbf{A}\mathbf{r}_k + \frac{\mu_k}{\mu_{k-1}}\tau_{k-1}\mathbf{r}_{k-1}.$$

CG is an efficient algorithm. The algorithmic complexity of CG is $\mathcal{O}(kn^2)$ for an $n \times n$ dense linear system that converges after k iterations. During each iteration of CG, matrix-vector multiplication is performed which requires $2n^2 - n$ operations, whereas the other vector operations are all $\mathcal{O}(n)$. This fact means that as the dimension of the system increases, the computational efficiency may be bounded by the number of times that matrix vector multiplication is performed, which is the number of iterations before converging to a desired tolerance. If the matrix is sparse, then the cost of matrix-vector multiplication

may be cheaper than $2n^2 - n$ operations. Further, the $n \times n$ matrix may not need to be stored in memory for the matrix multiplication to be calculated, which means CG would be even more efficient. Further, CG's memory requirements in Algorithm 3 include only three vectors: $\mathbf{r}_{k+1}$, $\mathbf{x}_{k+1}$, and $\mathbf{p}_{k+1}$. Otherwise, all of the computations can be performed efficiently in Algorithm 3 through the use of a few extra scalars in memory so that the amount of memory required by the algorithm grows relatively little as the dimensions of the system increase.

CG is also guaranteed to converge to the solution $\mathbf{x}$ by the nth iteration in exact arithmetic. The only reason that CG would be unable to converge is if the recurrence coefficients μ_k or τ_k were zero [95]. Since $\mathbf{A}$ is HPD, the only way that the coefficients are zero is if $\mathbf{r}_k$ is $\mathbf{0}$, which means the algorithm has converged. Equivalently for HPD matrices, there exists a $k+1$st vector in $\mathcal{K}_{k+1}(\mathbf{A}, \mathbf{b})$ that is linearly independent to $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ unless $\mathbf{x}$ is already an element of $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ [16].

CG can be given a tolerance, ϵ, as an input to return an approximation of $\mathbf{x}$ with at least a desired amount of accuracy. Choosing a larger tolerance often means that CG will converge to an approximation of $\mathbf{x}$ in fewer iterations. Let $\kappa(\mathbf{A}) = \|A\|_2 \|A^{-1}\|_2$ be the condition number of the matrix $\mathbf{A}$. For CG,

$$\begin{aligned} \frac{\|\mathbf{e}_k\|_{\mathbf{A}}}{\|\mathbf{e}_0\|_{\mathbf{A}}} &\leq 2 \left(\frac{\sqrt{\kappa(\mathbf{A})} - 1}{\sqrt{\kappa(\mathbf{A})} + 1} \right)^k, \\ \|\mathbf{e}_k\|_{\mathbf{A}} &\leq 2 \exp \left(\frac{-2k}{\sqrt{\kappa(\mathbf{A})}} \right) \|\mathbf{e}_0\|_{\mathbf{A}}. \end{aligned}$$

The first inequality can be found in Chapter 10 of Hackbusch's *Iterative Solution of Large Sparse Systems of Equations* [65], and the second inequality can be found on the conjugate gradient method page of Wikipedia [118]. The connection between the two inequalities can be proven using the series expansions of $\log(\sqrt{\kappa(\mathbf{A})} - 1) - \log(\sqrt{\kappa(\mathbf{A})} + 1)$ at infinity that yields $-2/\sqrt{(}\kappa) - \mathcal{O}(1/\sqrt{\kappa}^3)$, taking the exponential, and then raising the result to the kth

power. Further, using the relationship that $\mathbf{r}_k = \mathbf{A}\mathbf{e}_k$, we also have that

$$\|\mathbf{r}_k\|_2 \leq 2\exp\left(\frac{-2k}{\sqrt{\kappa(\mathbf{A})}}\right)\|\mathbf{A}^{1/2}\|_2\|\mathbf{e}_0\|_{\mathbf{A}}.$$

Therefore, when $k \geq \frac{1}{2}\sqrt{\kappa(\mathbf{A})}\log(2\|\mathbf{e}_0\|_{\mathbf{A}}\epsilon^{-1})$, the error at iteration k is less than or equal to ϵ; and when $k \geq \frac{1}{2}\sqrt{\kappa(\mathbf{A})}\log(2\|A^{1/2}\|_2\|\mathbf{e}_0\|_{\mathbf{A}}\epsilon^{-1})$, the residual at iteration k is less than or equal to ϵ.

Notice that the convergence bound of CG depends only on the condition number of the matrix, not directly on the dimension of the matrix. In most cases $\kappa(\mathbf{A})$ will increase as the dimension of $\mathbf{A}$ increases, but not always. The matrix representing the discrete potential equation for a graph is an example of a class of matrices where the condition number is bounded by a constant as the dimension increases [79]. There are many examples of matrices where CG converges in $k \ll n$ iterations [109, Lecture 38],[95]. These quickly converging examples usually have a small condition number, and either the eigenvalues are in small cluster(s) or all of the eigenvalues are relatively far from zero. The eigenvalues of the matrix representing the discrete potential equation for a graph are in a small cluster, and the rate of convergence is bounded by the minimum distance from any eigenvalue to the origin. Section 2.2 discusses how convergence can be further sped up through the use of a preconditioner, especially when $\kappa(\mathbf{A})$ is large.

While CG converges within n iterations in exact arithmetic, it is important to note that this is not always true in finite precision arithmetic[103, 43]. However, using finite precision arithmetic still leads to trustworthy approximations for $\mathbf{x}$. Chapter 4 of Greenbaum (1997) [57] goes through an error analysis of CG in finite precision arithmetic, and a convergence analysis can be found in both Paige (1980) [87] and Strakoš (1991) [103]. One conclusion from this research is that for matrices with a reasonably small condition number, CG converges quickly to good approximations. For matrices with a large condition number, it is possible to use a preconditioner and solve an equivalent system with a smaller condition number. Preconditioning is discussed more in Section 2.2.

In both exact and finite precision arithmetic, CG may be considered an approximation method. As long as the tolerance ϵ is greater than zero, CG may return an approximation $\mathbf{x}_k$ before finding the exact solution, which often allows CG to reach a desired tolerance in $k \ll n$ iterations. Since the 1970's, CG has been the method of choice for large HPD matrices. CG remains popular due to its computational efficiency, guaranteed lack of breakdown, and relatively few iterations to find a suitable approximate solution.

While CG is a powerful method, its largest drawback is that CG requires an HPD matrix as the input, and there are many problems where the matrix is not positive definite and/or not Hermitian. These non-Hermitian problems naturally occur in engineering, mathematical physics, data science, neuroscience, and other fields. In section 10 of their paper introducing CG, Hestenes and Steiffel point out that even if a matrix $\mathbf{A}$ is not HPD, if there exists an HPD matrix $\mathbf{M}$ such that $\mathbf{M}^{-1}\mathbf{A} = \mathbf{Y}$ and $\mathbf{Y}$ is HPD, then CG can still be used to solve the system of linear equations. In response, two major research threads took shape during the subsequent three decades: one was finding a Krylov subspace iterative algorithm that could be used to approximate solutions for matrices that are not HPD, and another was finding ways to characterize which systems could be transformed to Hermitian systems like Hestenes and Stiefel described so that CG would converge and finding what those transformations were. This section focuses on Krylov subspace methods for non-Hermitian matrices, and Section 2.2 will explore coordinate transformations that guarantee CG will converge for unexpected problems.

The goal of developing new iterative methods for non-HPD systems was to maintain as many of CG's positive qualities as possible. For example, GMRES was developed to allow any nonsingular matrix as an input while guaranteeing a lack of breakdown and minimizing the 2-norm of the residual restricted to $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$. However, GMRES is no longer a 3-term recurrence method so that the memory requirements grow during each iteration and GMRES is less computationally efficient than CG. On the other hand, Bi-CGSTAB is a 3-term recurrence method that no longer guarantees convergence. Krylov subspace methods

for non-Hermitian problems are faced with a trade-off between properties of CG to maintain: fewer computations per iteration, using less memory, monotonic convergence of the error and/or residual, guaranteed convergence, and convergence in $k \ll n$ iterations.

Algorithm 4 Generalized Minimum Residual Algorithm

Inputs: $\mathbf{A} \in \mathbb{C}^{n \times n}$, $\mathbf{b} \in \mathbb{C}^n$, max iterations m, and the initial guess $\mathbf{x}_0 \in \mathbb{C}^n$
$\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$; $\beta = \|\mathbf{r}_0\|_2$; $\mathbf{q}_1 = \mathbf{r}_0/\beta$
for $j = 1, 2, \cdots, m$ **do**
    $\tilde{\mathbf{q}}_{j+1} = \mathbf{A}\mathbf{q}_j$
    for $i = 1, 2, \cdots, j$ **do**
        $\mathbf{H}(i, j) = \langle \mathbf{q}_i, \tilde{\mathbf{q}}_{j+1} \rangle$
        $\tilde{\mathbf{q}}_{j+1} \leftarrow \tilde{\mathbf{q}}_{j+1} - \mathbf{H}(i, j)\mathbf{q}_i$
    end for
    $\mathbf{H}(j + 1, j) = \langle \tilde{\mathbf{q}}_{j+1}, \tilde{\mathbf{q}}_{j+1} \rangle^{1/2}$
    $\mathbf{q}_{j+1} = \tilde{\mathbf{q}}_{j+1}/\mathbf{H}(j + 1, j)$
end for
Define the $(m + 1) \times m$ Upper Hessenberg coefficient matrix $\mathbf{H}_{m+1,m} = \{\mathbf{H}(i, j)\}_{1 \le i \le m+1, 1 \le j \le m}$; Define the orthogonal basis of $\mathcal{K}_{m+1}(\mathbf{A}, \mathbf{b})$ $\mathbf{Q}_{m+1} = [\mathbf{q}_1 \cdots \mathbf{q}_{m+1}]$
Solve $\mathbf{y}_m = \text{argmin}_{\mathbf{y}} \|\beta \mathbf{e}_1 - \mathbf{H}_{m+1,m}\mathbf{y}\|_2$
Return: $\mathbf{x}_m = \mathbf{x}_0 + \mathbf{Q}_m \mathbf{y}_m$; $\beta_m = \|\mathbf{b} - \mathbf{A}\mathbf{x}_m\|_2$

GMRES was first introduced in 1986 by Saad and Schultz to compute solutions to systems of linear equations with non-singular non-Hermitian matrices[96]. GMRES is guaranteed to converge within n iterations for consistent linear systems [18] and the residual monotonically convergences in the 2-norm, including for highly non-normal matrices [96]. Unlike CG, which is derived from the Lanczos algorithm, GMRES directly uses the Arnoldi algorithm and then minimizes the residual. There are multiple ways to implement GMRES [123]. In Algorithm 4, modified Gram-Schmidt orthogonalization is used to orthogonalize $\mathbf{Q}_k$ in the Arnoldi algorithm. Instead, one could implement Arnoldi with Householder transformations [117]. The version of GMRES shown in Algorithm 4 also assumes that the number of iterations is known before running the algorithm and only minimizes the residual and solves for $\mathbf{x}_m$ after all m Arnoldi iterations are completed. GMRES can be restarted, which can be useful if the residual is not within the desired tolerance after m iterations.

Restarting GMRES means that convergence is no longer guaranteed within n iterations though. Alternatively, the GMRES algorithm can be modified to check the 2-norm of the residual during each iteration.

Regardless of which implementation is used, GMRES is less computationally efficient than CG. Both GMRES and CG have $\mathcal{O}(kn^2)$ arithmetic complexity for dense matrices, where k is the total number of iterations and n is the dimension of the linear system. The arithmetic complexity is the same for both algorithms because the matrix-vector multiplication is still the most costly operation during each iteration. A major difference between the two algorithms computationally is that GMRES uses a k-term recurrence instead of a 3-term recurrence to orthogonalize the $(k+1)$st basis vector of $\mathcal{K}_{k+1}(\mathbf{A}, \mathbf{b})$. This k-term recurrence results from the lack of complex conjugate symmetry in the input matrix. The growing recurrence relation linearly increases both the computation and memory used by GMRES. For very large systems with millions of dimensions, adding a new dimension n vector to accessible memory during each iteration quickly becomes cost prohibitive and may make GMRES unable to converge to within a specific error tolerance for certain problems. However, GMRES is still computationally feasible for larger systems than a direct method like Gaussian Elimination, which has an arithmetic complexity of $\mathcal{O}(n^3)$ for dense matrices, as long as $k \ll n$.

For all consistent linear systems, where $\mathbf{b}$ is in the range of $\mathbf{A}$, GMRES is guaranteed to converge within n iterations. GMRES uses the basis of $\mathcal{K}_k(\mathbf{A}, \mathbf{b}) = \text{span}\{\mathbf{b}, \mathbf{Ab}, \cdots, \mathbf{A}^{k-1}\mathbf{b}\}$ generated by the Arnoldi algorithm to form $\mathbf{x}_k$, and by the nth iteration Arnoldi forms a basis spanning the range of $\mathbf{A}$. When $\mathbf{A}$ is non-singular, a basis of the range of $\mathbf{A}$ is formed during the nth iteration. When $\mathbf{A}$ is singular, if $\mathbf{A}$ is diagonalizable and $\mathbf{b}$ has nonzero components in the direction of each eigenvector of $\mathbf{A}$, then a basis of the range of $\mathbf{A}$ is formed before the nth iteration[2]. Once the basis is formed, any vector $\mathbf{b}$ in the range of $\mathbf{A}$ can be written as a weighted sum of the basis vectors, guaranteeing the convergence of

[2]For a proof of the convergence of consistent linear systems with a singular matrix, see Brown and Walker (1997)[18].

GMRES in n or fewer iterations since GMRES finds the optimal approximation from this space. GMRES may converge to an exact solution in fewer than n iterations when $\mathbf{b}$ is deficient in certain eigencomponents, but GMRES will never require greater than n iterations to find the solution for consistent linear systems.

Due to the linearly growing memory costs during each iteration of GMRES, one would ideally be able to choose a maximum iteration $m \ll n$ for Algorithm 4 that will simultaneously converge and avoid unnecessary computations. There are convergence bounds for GMRES that we can calculate for any linear system. Let $\mathcal{P}_k$ be the set of all polynomials with degree k or less and with a value of 1 at the origin of the complex plane, and assume that $\mathbf{A}$ is diagonalizable with eigendecomposition $\mathbf{A} = \mathbf{V}\Lambda\mathbf{V}^{-1}$. Then

$$\|\mathbf{r}_k\|_2 = \min_{p_k \in \mathcal{P}_k} \|\mathbf{V} p_k(\Lambda) \mathbf{V}^{-1} \mathbf{r}_0\|_2. \tag{2.1}$$

Equation (2.1) then implies

$$\frac{\|\mathbf{r}_k\|_2}{\|\mathbf{r}_0\|_2} \leq \kappa(V) \min_{p_k \in \mathcal{P}_k} \max_{i \in 1,2,\cdots,n} |p_k(\lambda_i)|. \tag{2.2}$$

However, for any given non-increasing convergence curve there exists a matrix $\mathbf{A}$ and right hand side $\mathbf{r}_0$ that follows that convergence curve where $\mathbf{A}$ may have any set of eigenvalues we choose [59]. This result implies that eigenvalues may not yield enough information to tell one about the convergence of GMRES in a sharp way. If the matrix $\mathbf{A}$ is normal or close to normal, where $\kappa(\mathbf{V})$ is near 1, then equation (2.2) gives a sharp or reasonable approximation of the convergence, respectively. When $\mathbf{A}$ is highly non-normal, equation (2.2) yields little information about the convergence.

An alternative bound to equation (2.1), assuming that the numerical range of the matrix $\mathbf{A}$, $W(\mathbf{A})$, does not contain the origin [30], is

$$\frac{\|\mathbf{r}_k\|_2}{\|\mathbf{r}_0\|_2} \leq (1 + \sqrt{2}) \min_{p_k \in \mathcal{P}_k} \max_{z \in W(\mathbf{A})} |p_k(z)|_2. \tag{2.3}$$

Improving the convergence bounds for highly non-normal matrices is an open field of research, and my dissertation discusses this topic in-depth in Chapters 4 and 5.

The GMRES convergence bounds (2.2) and (2.3) may be large overestimates of the 2-norm of the residual. As Algorithm 4 is written, GMRES will run for m iterations, regardless of converging before the mth iteration or not. So even when m is chosen to satisfy these theoretical convergence bounds, we may still be performing more computations than necessary to approximate $\mathbf{x}$ to within the desired tolerance. There are two common modifications to GMRES to deal with choosing the number of iterations: Restarted GMRES and GMRES with Givens rotations which is shown in Algorithm 5.

Restarted GMRES, GMRES(m), is a useful way to limit the amount of memory used while approximating $\mathbf{x}_k$. GMRES(m) runs Arnoldi for m iterations and then calculates the 2-norm of the residual. If the residual is within a given tolerance bound, then the approximation $\mathbf{x}_m$ is returned by the algorithm. If the residual is not within the given tolerance bound, Arnoldi is run for another m iterations this time with the returned vector $\mathbf{x}_m$ from the previous call of GMRES(m) used as the input vector $\mathbf{x}_0$ and $\mathbf{q}_1 = \mathbf{r}_m/\|\mathbf{r}_m\|_2$ where $\mathbf{r}_m$ is also from the previous call of GMRES(m). While Restarted GMRES ensures the memory costs stay below $m+3$ vectors and an $m \times m$ upper Hessenberg matrix, restarting GMRES means that we lose the guarantee that $\mathbf{x}$ will be computed within n iterations, and except in special cases, there are no convergence bounds for GMRES(m)[77]. Choosing an appropriate m that is not so small that GMRES(m) converges very slowly or fails to converge at all while keeping the computational costs of each run of Arnoldi relatively small is a difficult problem, and there are examples where the GMRES residuals stagnate and do not converge before this point [3, 77]

An alternative modification to Algorithm 4 allows us to calculate $\|\mathbf{r}_j\|_2$ efficiently during each iteration. Algorithm 5 shows GMRES implemented with Givens rotations, which are used to convert the upper Hessenberg coefficient matrix output by the Arnoldi algorithm into an upper triangular coefficient matrix, and used to calculate the 2-norm of the residual

Algorithm 5 Generalized Minimum Residual Algorithm with Givens Rotations

Inputs: $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\mathbf{b} \in \mathbb{C}^n$, tolerance ϵ, max iterations m, and the initial guess $\mathbf{x}_0 \in \mathbb{C}^n$
$\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$; $\mathbf{g}(1) = \|\mathbf{r}_0\|_2$
$\mathbf{q}_1 = \mathbf{r}_0/\mathbf{g}(1)$
$j = 0$.
while $|\mathbf{g}(j+1)|/\|\mathbf{b}\|_2 > \epsilon$ and $j < m$ **do**
    $j = j + 1$
    $\tilde{\mathbf{q}}_{j+1} = \mathbf{A}\mathbf{q}_j$
    for $i = 1, 2, \cdots, j$ **do**
        $\mathbf{U}(i,j) = \langle \mathbf{q}_i, \tilde{\mathbf{q}}_{j+1} \rangle$
        $\tilde{\mathbf{q}}_{j+1} \leftarrow \tilde{\mathbf{q}}_{j+1} - \mathbf{U}(i,j)\mathbf{q}_i$
    end for
    $\mathbf{U}(j+1,j) = \|\tilde{\mathbf{q}}_{j+1}\|_2$
    $\mathbf{q}_{j+1} = \tilde{\mathbf{q}}_{j+1}/\mathbf{U}(j+1,j)$
    for $i = 1, 2, \cdots j-1$ **do**
        $t = \mathbf{c}(i)\mathbf{U}(i,j) + \mathbf{s}(i)\mathbf{U}(i+1,j)$
        $\mathbf{U}(i+1,j) \leftarrow \mathbf{c}(i)\mathbf{U}(i+1,j) - \bar{\mathbf{s}}(i)\mathbf{U}(i,j)$
        $\mathbf{U}(i,j) \leftarrow t$
    end for
    Compute the jth rotation, $\mathbf{c}(j)$ and $\mathbf{s}(j)$, to annihilate $\mathbf{U}(j+1,j)$.
    $\mathbf{U}(j,j) \leftarrow \mathbf{c}(j)\mathbf{U}(j,j) + \mathbf{s}(j)\mathbf{U}(j+1,j)$
    $\mathbf{U}(j+1,j) \leftarrow 0$
    $\mathbf{g}(j+1) = -\bar{\mathbf{s}}(j)\mathbf{g}(j)$
    $\mathbf{g}(j) \leftarrow \mathbf{c}(j)\mathbf{g}(j)$
end while
Define the upper triangular coefficient matrix $\mathbf{U}_j = \{\mathbf{U}(i,k)\}_{1\le i\le j, 1\le k\le j}$
Define $\mathbf{Q}_j = [\mathbf{q}_1\mathbf{q}_2\cdots\mathbf{q}_j]$
Solve $\mathbf{U}_j\mathbf{y}_j = \mathbf{g}(1:j)$
Return: $\mathbf{x}_j = \mathbf{x}_0 + \mathbf{Q}_j\mathbf{y}_j$

vector during each iteration. Calculating the 2-norm of the residual during each iteration allows one to use a while loop that stops running once the algorithm achieves the desired tolerance, similar to the while loop in Algorithm 3 for CG. Since we have converted the upper Hessenberg matrix into an upper triangular one, the minimal $\mathbf{y}_j$ can be found with back substitution, which for a dense matrix has an arithmetic complexity of $\mathcal{O}(n^2)$. Overall, GMRES with Givens rotations has an arithmetic complexity of $\mathcal{O}(kn^2)$ for a dense matrix (so worst case), once again because the matrix-vector multiplication is the most costly operation in the algorithm. Despite having the same arithmetic complexity, including Givens rotations in the GMRES algorithm does increase the number of computations necessary per iteration, but reduces the computations necessary to compute the residual during each iteration. Also, the memory requirements for GMRES with Givens rotations in algorithm 5 increases by two vectors that contain the cosines and sines of the rotations calculated during each rotation. Especially for highly non-normal $\mathbf{A}$, where the convergence bound (2.2) is likely to be a large overestimate, the Givens Rotation version of the algorithm can still be more efficient than the version of GMRES in Algorithm 4 by reducing the number of iterations used. Using GMRES with Givens rotations like in Algorithm 5 is useful when one wishes to maintain the guaranteed convergence of the full GMRES algorithm while simultaneously minimizing the iterations used to reach a desired tolerance, which keeps the memory used by the algorithm to a minimum.

An alternative iterative method to GMRES for non-Hermitian matrices is Bi-CGSTAB. Its strengths and weaknesses differ from those of GMRES. Bi-CGSTAB's strengths are that it uses a constant amount of memory, storing five n-dimensional vectors during each iteration, and converges relatively quickly when it converges. Bi-CGSTAB achieves this constant memory usage by forming residual and search directions equivalent to those from the biconjugate gradient (BiCG) algorithm. The BiCG algorithm forms biorthogonal search directions $\mathbf{p}_k$ and $\hat{\mathbf{p}}_k$ from $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ and $\mathcal{K}_k(\mathbf{A}^*, \mathbf{b})$ respectively, where for all $i \neq j$ $\langle \hat{\mathbf{p}}_i, \mathbf{A}\mathbf{p}_j \rangle = 0$. These search directions are then used to compute the biorthogonal residuals $\mathbf{r}_k = \mathbf{b} - \mathbf{A}\mathbf{x}_k$

Algorithm 6 Bi-Conjugate Gradient Stabilized Algorithm

Inputs: $\mathbf{A} \in \mathbb{C}^{n \times n}$, $\mathbf{b} \in \mathbb{C}^n$, tolerance ϵ, and initial guess $\mathbf{x}_0 \in \mathbb{C}^n$
$\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$
Let $\tilde{\mathbf{r}}_0$ be an arbitrary vector where $\langle \tilde{\mathbf{r}}_0, \mathbf{r}_0 \rangle \neq 0$ and $\|\tilde{\mathbf{r}}_0\|_2 = 1$.
$\mathbf{p}_0 = \mathbf{r}_0$
$j = 0$
while $\|\mathbf{r}_j\|_2 > \epsilon$ and $j < m$ **do**
    $\alpha_j = \langle \tilde{\mathbf{r}}_0, \mathbf{r}_j \rangle / \langle \tilde{\mathbf{r}}_0, \mathbf{A}\mathbf{p}_j \rangle$
    $\mathbf{w}_j = \mathbf{r}_j - \alpha_j \mathbf{A}\mathbf{p}_j$
    $\omega_{j+1} = \langle \mathbf{A}\mathbf{w}_j, \mathbf{w}_j \rangle / \langle \mathbf{A}\mathbf{w}_j, \mathbf{A}\mathbf{w}_j \rangle$
    $\mathbf{x}_{j+1} = \mathbf{x}_j + \alpha_j \mathbf{p}_j + \omega_{j+1}\mathbf{w}_j$
    $\mathbf{r}_{j+1} = \mathbf{w}_j - \omega_{j+1}\mathbf{A}\mathbf{w}_j$
    $\beta_{j+1} = (\alpha_j / \omega_{j+1}) \ \langle \tilde{\mathbf{r}}_0, \mathbf{r}_{j+1} \rangle / \langle \tilde{\mathbf{r}}_0, \mathbf{r}_j \rangle$
    $\mathbf{p}_{j+1} = \mathbf{r}_{j+1} + \beta_{j+1}(\mathbf{p}_j - \omega_{j+1}\mathbf{A}\mathbf{p}_j)$
    $j \leftarrow j + 1$
end while
Return: $\mathbf{x}_j$

and $\hat{\mathbf{r}}_k = \mathbf{b} - \mathbf{A}^*\mathbf{x}_k$ respectively, which satisfy that for all $i \neq j$ $\langle \mathbf{r}_i, \hat{\mathbf{r}}_j \rangle = 0$. Where BiCG explicitly forms two 3-term recurrences, Bi-CGSTAB implicitly solves for the same search directions and residual directions for the system $\mathbf{A}\mathbf{x} = \mathbf{b}$ without calculating the biorthogonal ones associated with $\mathbf{A}^*\mathbf{x} = \mathbf{b}$.

Using a similar amount of work as BiCG, Bi-CGSTAB instead speeds up and smooths the convergence of the algorithm. This relatively fast convergence and the limited amount of memory used by Bi-CGSTAB are two strengths of the algorithm. In fact, the memory used by Bi-CGSTAB every iteration is similar to the memory used during only the second iteration of GMRES, which means Bi-CGSTAB uses much less memory than GMRES.

A potential weakness of Bi-CGSTAB is that it performs two matrix-vector products per iteration. For a very large dense matrix $\mathbf{A}$, the computational cost of performing a second matrix-vector product during each iteration, and the resulting increase in running time, may make GMRES a more reasonable algorithmic choice.

Another weakness of Bi-CGSTAB is that the algorithm is not guaranteed to converge for every linear system. In practice, Bi-CGSTAB converges frequently enough that many will

use it to approximate solutions to systems of linear equations without any modifications. However, in a 1995 paper by Brezinski and Redivozaglia, there are two example matrices where Bi-CGSTAB does not converge [16], and occasionally practitioners will come across other matrices that don't converge. Bi-CGSTAB does not minimize the residual or error from each iteration in a norm of interest; which means that the residuals from Bi-CGSTAB are not orthogonal in the 2-norm and that the 2-norm of the residual and error may increase between iterations [112]. In contrast, CG minimizes the **A**-norm of the error during each iteration, and GMRES minimizes the 2-norm of the residual during each iteration. This minimization in norms of interest leads to a monotonic decrease in the **A**-norm of the error and the 2-norm of the residual for CG and GMRES, respectively.

There are many more Krylov subspace algorithms than the three discussed in this chapter. Some Krylov subspace algorithms are optimal for solving linear systems of equations for specific classes of matrices like CG for HPD matrices, and MINRES for indefinite Hermitian matrices[3]. Different Krylov subspace methods will be more or less appropriate depending on the linear system of equations one is trying to solve and the goals of solving that system. For example, GMRES guarantees convergence, but Bi-CGSTAB uses less memory than GMRES. For non-Hermitian problems, different Krylov subspace algorithms have different strengths and weaknesses: computational efficiency per iteration such as using a single matrix-vector product, memory requirements, monotonic convergence of the error and/or residual in a norm of interest such as the 2-norm, guaranteed convergence within n iterations, and frequent convergence in $j \ll n$ iterations in practice.

2.2 Preconditioners

When computing solutions to $\mathbf{Ax} = \mathbf{b}$, Preconditioners are commonly used to reduce the condition number of the matrix, $\kappa(\mathbf{A})$, to a value close to one. While discussing CG, we

[3]MINRES is not used or introduced in this dissertation. Instead, readers are directed to either the original paper [88] or Saad's or Greenbaum's texts [95] and [57] if they are interested in learning more about MINRES.

saw that the convergence bound may be written such that it only depends on $\kappa(\mathbf{A})$. This dependence implies that reducing the condition number of the system being solved may reduce the number of iterations necessary for convergence. While discussing GMRES, we saw that the bound (2.2), based on the eigenvalues of $\mathbf{A}$, is only reasonable when $\kappa(\mathbf{V}) \approx 1$, where $\mathbf{A} = \mathbf{V}\Lambda\mathbf{V}^{-1}$ is an eigenvalue decomposition of $\mathbf{A}$. For GMRES, a preconditioner may both decrease the number of iterations necessary for convergence and reduce the problem of bounding convergence from finding the maximum of a polynomial in a region of the complex plane such as the numerical range, see bound (2.3), to finding the maximum of a polynomial over a finite set of complex numbers in a subset of the complex plane such as the eigenvalues, see bound (2.2).

A preconditioner matrix transforms a linear system into an equivalent system that is ideally somehow more efficient to solve. Let $\mathbf{M} = \mathbf{C}\mathbf{C}^*$, where $\mathbf{M}$ is Hermitian positive definite (HPD). Then we may use $\mathbf{M}$ or $\mathbf{C}$ as the preconditioner for the linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ to get

$$\mathbf{M}^{-1}\mathbf{A}\mathbf{x} = \mathbf{M}^{-1}\mathbf{b} \quad \text{or} \tag{2.4}$$

$$(\mathbf{C}^{-1}\mathbf{A}\mathbf{C}^{-*})(\mathbf{C}^*\mathbf{x}) = \mathbf{C}^{-1}\mathbf{b}. \tag{2.5}$$

Equations 2.4 and 2.5 are known as the left and symmetric preconditioned systems, respectively. Solving equation (2.4) with a Krylov subspace method computes $\mathbf{x}_k$'s that approximate solutions to the original linear system, whereas solving equation (2.5) with a Krylov subspace method computes $\mathbf{y}_k = \mathbf{C}^*\mathbf{x}_k$. In equation (2.5), $\mathbf{C}^{-1}\mathbf{A}\mathbf{C}^{-*}$ is a $*$-congruence transformation that maintains the Hermitian properties of the matrix $\mathbf{A}$. One reason to use (2.5) to precondition a system, despite the extra matrix-vector multiplication, is if $\mathbf{A}$ is already Hermitian and we want to use an algorithm like CG that requires a Hermitian matrix input to maintain guaranteed convergence. Congruence transformations also maintain the signs, though not values, of the eigenvalues so that if $\mathbf{A}$ is HPD then $\mathbf{C}^{-1}\mathbf{A}\mathbf{C}^{-*}$ is HPD [104].

For most Krylov subspace algorithms there is a preconditioned version that efficiently accounts for the extra matrix-vector multiplications that using a preconditioner requires. For example, preconditioned conjugate gradient (PCG) is shown in Algorithm 7.

Algorithm 7 Preconditioned Conjugate Gradient Algorithm

Inputs: $\mathbf{A} \in \mathbb{C}^{n \times n}$, $\mathbf{b} \in \mathbb{C}^n$, tolerance ϵ, initial guess $\mathbf{x}_0 \in \mathbb{C}^n$, maximum iterations m, and Hermitian positive definite preconditioner $\mathbf{M}$. Either $\mathbf{A}$ or $\mathbf{M}^{-1}\mathbf{A}$ must be HPD.
$\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$
Solve $\mathbf{M}\mathbf{z}_0 = \mathbf{r}_0$
$\mathbf{q}_0 = \mathbf{z}_0$
$k = 0$
while $\|\mathbf{r}_k\|_2 > \epsilon$ and $k \leq m$ **do**
  $\alpha_k = \langle \mathbf{r}_k, \mathbf{z}_k \rangle / \langle \mathbf{A}\mathbf{q}_k, \mathbf{q}_k \rangle$
  $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{q}_k$
  $\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{q}_k$
  Solve $\mathbf{M}\mathbf{z}_{k+1} = \mathbf{r}_{k+1}$
  $\beta_k = \langle \mathbf{r}_{k+1}, \mathbf{z}_{k+1} \rangle / \langle \mathbf{r}_k, \mathbf{z}_k \rangle$
  $\mathbf{q}_{k+1} = \mathbf{z}_{k+1} + \beta_k \mathbf{q}_k$
  $k \leftarrow k + 1$
end while

Examples of preconditioners that are used to reduce the condition number of a system include diagonal preconditioners for diagonally dominant matrices, incomplete Cholesky factorization preconditioners for HPD matrices [5], domain decomposition preconditioners such as analytic incomplete LU and balancing domain decomposition by constraints used to solve elliptical partial differential equations [51, 32], and preconditioners designed for a specific problem such as diffusion synthetic acceleration (DSA) preconditioners for solving the S_N transport equations [86, 99]. Using preconditioners to reduce the condition number of the linear system is justified when (cost of computing the preconditioner) + (cost per iteration with the preconditioned system)*(number of iterations to convergence of the preconditioned system) < (cost per iteration)*(number of iterations to convergence without the preconditioner). For a preconditioned system, the computational cost per iteration will usually increase, and the amount it increases depends on how efficiently matrix-vector multiplication can be applied with the preconditioner matrix and if a change of coordinates

is necessary to compute the approximate solution to the original system (where both often mean solving a triangular linear system). Usually a preconditioner must be computed for each matrix individually. However, if we already know a preconditioner matrix for which the ratio of the largest to smallest eigenvalue of $\mathbf{M}^{-1}\mathbf{A}$ is $\ll \kappa(\mathbf{A})$, then we only need the reduction in iterations to outweigh the increased computational cost per iteration.

For PCG, the preconditioner must be HPD to maintain guaranteed convergence. For a general matrix $\mathbf{C}$, let $\mathbf{M} = \mathbf{C}\mathbf{C}^*$. Then $\mathbf{M}$ is HPD as long as $\mathbf{C}$ is not singular. This fact implies that $\mathbf{C}$ can have any structure and even be highly non-normal. For an HPD matrix $\mathbf{A}$, a common algorithm for computing a preconditioner is the incomplete Cholesky factorization, which outputs $\mathbf{A} \approx \mathbf{L}\mathbf{L}^*$ where $\mathbf{L}$ is a sparse lower triangular matrix.

Preconditioners can also be used to transform the coordinate basis of a linear system to one that satisfies an algorithm's input requirements. For example, since CG uses relatively little memory, has guaranteed convergence, and is one of the fastest converging iterative methods [57], preconditioners that would guarantee systems of equations with non-Hermitian matrices converge for CG are interesting. The conjugate transpose of a matrix can always be used to transform a non-Hermitian linear system to an HPD linear system since $\mathbf{A}^*\mathbf{A}$ is HPD for all non-singular matrices $\mathbf{A}$. The conjugate gradient algorithm for the normal equations (CGNR) and (CGNE) are two algorithms to solve

$$\mathbf{A}^*\mathbf{A}\mathbf{x} = \mathbf{A}^*\mathbf{b}$$

and

$$\mathbf{A}\mathbf{A}^*\mathbf{y} = \mathbf{b},\ \mathbf{x} = \mathbf{A}^*\mathbf{y}$$

respectively. A major concern when using either CGNR or CGNE is that the convergence rate of these algorithms is now bounded by $\kappa(\mathbf{A}^*\mathbf{A})$ and $\kappa(\mathbf{A}\mathbf{A}^*)$ respectively, which are both equal to $\kappa(\mathbf{A})^2$.

Alternatively, if we find HPD matrices $\mathbf{M}$ and $\mathbf{Y}$ such that $\mathbf{A} = \mathbf{M}\mathbf{Y}$, then the linear

system $\mathbf{Ax} = \mathbf{b}$ can be solved with CG by using $\mathbf{M}^{-1}$ as a left preconditioner like in equation (2.4), and CG is guaranteed to converge.

An example of where these preconditioners show up is when solving systems of linear equations involving the potential of a graph Laplacian matrix, $\mathbf{G} = \mathbf{D} - \mathbf{A}$, where $\mathbf{G} \in \mathbb{R}^{n \times n}$, $\mathbf{D}$ is the degree matrix, and $\mathbf{A}$ is the adjacency matrix. In the graph Laplacian $\mathbf{G}$, each vertex i in a graph is represented by the ith row and ith column of the square matrix. The degree matrix, $\mathbf{D}$, is a diagonal matrix where the ith diagonal element contains the sum of edge weights pointing from vertex i in the graph. The adjacency matrix $\mathbf{A}$ contains the edge weights of the graph, where the entry in row i and column j is equal to the edge weight of the edge pointing from vertex i to vertex j in the graph. This definition means that the diagonal entries in $\mathbf{D}$ are equivalent to the respective row sums of $\mathbf{A}$. Thus, each row sum of $\mathbf{G}$ is zero, and $\mathbf{G}$ is a singular and positive semi-definite matrix. Further, if $\mathbf{G}$ represents an undirected graph, such that for every edge pointing from vertex i to vertex j there is an edge with equal weight pointing from vertex j to vertex i, then $\mathbf{G}$ is also symmetric.

First, to address the issue that $\mathbf{G}$ is singular, we introduce a constant α with $0 < \alpha \leq 1$ to get the matrix $\mathbf{G}_\alpha = \mathbf{D} - \alpha\mathbf{A}$. Notice that $\mathbf{G} = \lim_{\alpha \to 1} \mathbf{D} - \alpha\mathbf{A}$, but that for all other values of α the matrix is now positive definite. In some applications, α is known as a discounting factor [85]. We assume that $0 < \alpha < 1$ for the rest of this section.

Depending on the magnitude of the edge weights, $\mathbf{D} - \alpha\mathbf{A}$ may have a rather large condition number. However, the matrix $\mathbf{D}^{-1}\mathbf{G}_\alpha = \mathbf{I} - \alpha\mathbf{P}$, where $\mathbf{P} = \mathbf{D}^{-1}\mathbf{A}$ satisfies the properties of a probability transition matrix of a discrete time Markov chain, is well-conditioned. One property of a probability transition matrix is that all of the eigenvalues are contained in the unit disk. Then, all of the eigenvalues of $\mathbf{I} - \alpha\mathbf{P}$ are contained in $\mathcal{D}(1, \alpha)$, $\mathbf{I} - \alpha\mathbf{P}$ is a positive definite system, and

$$\mathbf{D}^{-1}(\mathbf{D} - \alpha\mathbf{A})\mathbf{x} = \mathbf{b}, \tag{2.6}$$

has a unique solution $\mathbf{x}$. The linear system in (2.6) is known as the discrete potential

equation (DPE), and one popular use for the DPE is as a local proximity measure between vertices in a graph [85, Ch 4.2].

In most cases, $\mathbf{I} - \alpha\mathbf{P}$ is not a symmetric system. However, when $\mathbf{D} - \alpha\mathbf{A}$ is symmetric, then PCG is guaranteed to converge for the system in (2.6) when $\mathbf{M} = \mathbf{D}$ is the preconditioner in Algorithm 7. For graph Laplacian problems, using $\mathbf{D}$ as the preconditioner allows us to take advantage of the symmetry of $\mathbf{G}$ and take advantage of the positive definite properties of $\mathbf{I} - \alpha\mathbf{P}$.

It is also possible to find a preconditioner for a graph Laplacian that does not have a symmetric adjacency matrix when the decomposition $\mathbf{G} = \mathbf{MY}$ exists, where $\mathbf{M}$ and $\mathbf{Y}$ are both SPD[4]. This decomposition exists if and only if there exists a 3-term recurrence method to solve a linear system with $\mathbf{A}$ as the coefficient matrix, and may exist even when $\mathbf{A}$ is non-Hermitian and potentially highly non-normal. In 1984, Faber and Manteuffel proved the necessary and sufficient conditions for when an s-term recurrence exists [44]:

Theorem 2.1:
An s-term CG method exists for the matrix $\mathbf{A}$ if and only if either

1. the minimal polynomial of $\mathbf{A}$ has degree less than or equal to s, or

2. $\mathbf{A}^\dagger$ is a polynomial of degree less than or equal $s-2$ in $\mathbf{A}$, where $\mathbf{A}^\dagger$ is the adjoint of $\mathbf{A}$ with respect to some inner product, that is $\langle\langle \mathbf{Ax}, \mathbf{y}\rangle\rangle = \langle\langle \mathbf{x}, \mathbf{A}^\dagger\mathbf{y}\rangle\rangle$ for all vectors $\mathbf{x}$ and $\mathbf{y}$.

The first condition is only satisfied if there are three or fewer eigenvalues (though this is not a sufficient condition[5]). The first condition is not frequently checked though, because for Krylov subspace methods like GMRES, if a matrix satisfies the first condition then the method will converge in three or fewer iterations without modification. The second condition

[4]Note that we only need symmetry because $\mathbf{G}$ only has real entries.

[5]A 3-term recurrence method for solving a system of linear equations with the matrix $\mathbf{A}$ exists if for each distinct eigenvalue of $\mathbf{A}$ we sum the dimension of the largest Jordan block and the value of that sum is less then or equal to three.

states that a 3-term recurrence method exists for $\mathbf{A}$ when there is an inner product defined by an HPD matrix $\mathbf{W}$, where $\langle\langle \cdot, \cdot \rangle\rangle = \langle \cdot, \mathbf{W} \cdot \rangle$, such that the adjoint of $\mathbf{A}$ defined by that inner product, $\mathbf{A}^\dagger = \mathbf{W}^{-1}\mathbf{A}^*\mathbf{W}$, is a linear function of $\mathbf{A}$. In other words, $\mathbf{A}^\dagger = c_1\mathbf{A} + c_0\mathbf{I}$ where c_0 and c_1 are constants[6]. When such an inner product exists, this is also equivalent to saying that $\mathbf{A}$ is $\mathbf{W}$-normal and that $\mathbf{A}\mathbf{A}^\dagger = \mathbf{A}^\dagger\mathbf{A}$. The matrix $\mathbf{A}$ is only $\mathbf{W}$-normal when all of the eigenvalues lie in a straight line in the complex plane, such that the matrix $\mathbf{W}^{1/2}\mathbf{A}\mathbf{W}^{-1/2}$ is a shifted and rotated Hermitian matrix [44, 57].

In the graph Laplacian example, if $\mathbf{G}_\alpha$ is restricted to only real entries, then the decomposition $\mathbf{G}_\alpha = \mathbf{M}\mathbf{Y}$ with $\mathbf{M}$ SPD and $\mathbf{Y}$ SPD only exists when all of the eigenvalues of $\mathbf{G}_\alpha$ are real. In that case, CG is guaranteed to converge for the linear system $\mathbf{Y}\mathbf{x} = \mathbf{M}\mathbf{b}$. And, instead of explicitly forming $\mathbf{Y}$ or writing a separate function to calculate $\mathbf{M}^{-1}\mathbf{A}$ times a vector during each iteration, we may run PCG on the linear system $\mathbf{G}_\alpha\mathbf{x} = \mathbf{b}$ with $\mathbf{M}$ as the left preconditioner.

Beyond the graph Laplacian example, there also exist many highly non-normal matrices satisfying the conditions for a 3-term recurrence method to exist, for example the non-periodic Hatano-Nelson matrix on pg. 341 of Trefethen and Embree has all real eigenvalues [108]. The implication of Theorem 2.1 is that for any $\mathbf{A}$ with eigenvalues in a straight line, PCG can be used to approximate the solution of $\mathbf{A}\mathbf{x} = \mathbf{b}$ as long as a decomposition $\mathbf{A} = \mathbf{M}\mathbf{Y}$ is known. Further, when $\mathbf{A}$ has complex entries this straight line of eigenvalues is no longer restricted to the real axis. The difficulty in practice is finding an HPD matrix $\mathbf{M}$ such that $\mathbf{M}^{-1}\mathbf{A} = \mathbf{Y}$, where $\mathbf{M}$ and $\mathbf{Y}$ are HPD matrices. When all eigenvalues of $\mathbf{A}$ have positive real parts and $\mathbf{A}$ is diagonalizable such that $\mathbf{A} = \mathbf{V}\Lambda\mathbf{V}^{-1}$, one possibility is to define $\mathbf{M} = \mathbf{V}\mathbf{V}^*$ and $\mathbf{Y} = \mathbf{V}^{-*}\Lambda\mathbf{V}^{-1}$. Then $\mathbf{M}$ and $\mathbf{Y}$ satisfy $\mathbf{A} = \mathbf{M}\mathbf{Y}$ and are both HPD. However, the eigenvalues and eigenvectors are not usually known, and can be computationally expensive to compute. Then while $\mathbf{M}$ and $\mathbf{Y}$ can be defined using the eigenvalues and eigenvectors of $\mathbf{A}$, it is still an open problem to find a more efficient

[6]The matrix $\mathbf{W}$ defining this inner-product can be related back to the left preconditioner matrix $\mathbf{M}^{-1}$ by the relation $\mathbf{W} = \mathbf{M}^{-1}$.

decomposition of **A** for the purposes of using PCG when Theorem 2.1 is satisfied.

Using a preconditioned algorithm allows one to more efficiently approximate the solution to a linear system of equations. Preconditioners achieve this result by reducing the condition number of the problem and/or transforming the linear system to one for which a more efficient algorithm exists. When choosing a preconditioner, a major concern is keeping the resulting computational cost from overtaking any efficiency gained from using fewer iterations to solve the preconditioned system. Because of this concern, preconditioners are often calculated with algorithms specialized for different types of linear systems.

Chapter 3

THE ACTION OF RATIONAL FUNCTIONS OF A MATRIX ON A VECTOR

Computing the action of rational functions of a matrix on a vector is relevant to a variety of applications. Examples include the action of dynamical systems defined by the matrix exponential on a population, computing exponential time integrators to solve ordinary differential equations of the form $\frac{d}{dt}u(t) = F(u(t)),\ u(t_0) = u_0$, and computing trigonometric functions of a matrix, for example to solve second-order systems of differential equations. We focus on computing the action of rational functions of a matrix on a vector with the understanding that functions analytic in a region of the complex plan can be approximated arbitrarily well by rational functions within that region.

Let $\mathbf{A} \in \mathbb{C}^{n\times n}$ be non-Hermitian, and potentially highly non-normal. Let $R(\mathbf{A})$ be a rational function of $\mathbf{A}$, where $R(\mathbf{A}) = D(\mathbf{A})^{-1}N(\mathbf{A})$. The polynomial $N(z)$ has maximum degree ν, and represents the numerator polynomial of the rational function. The polynomial $D(z)$ has maximum degree μ, and represents the denominator polynomial of the rational function. We assume that $D(\mathbf{A})$ is non-singular. We refer to $\mathbf{x}$ as the result of applying the rational function of a matrix to a vector such that $\mathbf{x} = R(\mathbf{A})\mathbf{b}$.

Practitioners are known to sometimes compute $R(\mathbf{A})$, and then calculate the matrix-vector product $R(\mathbf{A})\mathbf{b}$ [46, 82, 66]. Our goal is to avoid calculating $R(\mathbf{A}), N(\mathbf{A})$, or $D(\mathbf{A})$ directly, and instead compute $\mathbf{x} = R(\mathbf{A})\mathbf{b}$ using only matrix-vector products. For example, if one were to use GMRES to solve $\mathbf{x} = R(\mathbf{A})\mathbf{b}$, one could use $D(\mathbf{A})$ as the input matrix and $N(\mathbf{A})\mathbf{b}$ as the right-hand side vector to solve the system $D(\mathbf{A})\mathbf{x} = N(\mathbf{A})\mathbf{b}$. To use the built-in GMRES algorithm that comes with many programming languages like Python and MATLAB and avoid forming the matrix $D(\mathbf{A})$ through matrix-matrix multiplication requires writing a function to compute $D(\mathbf{A})\mathbf{x}$ as a series of matrix-vector products using the

matrix $\mathbf{A}$. Otherwise, matrix-matrix multiplication with dense matrices has an algorithmic complexity of $O(n^3)$. And even if the matrix $\mathbf{A}$ is sparse, then $D(\mathbf{A})$ may be less sparse.

Applying GMRES to the system $D(\mathbf{A})\mathbf{x} = N(\mathbf{A})\mathbf{b}$ minimizes the residual norm

$$\begin{aligned} \|\mathbf{r}_k\|_2 =& \|N(\mathbf{A})\mathbf{b} - D(\mathbf{A})\mathbf{x}_k\|_2 && (3.1)\\ =& \|R(\mathbf{A})\mathbf{b} - \mathbf{x}_k\|_{D(\mathbf{A})^* D(\mathbf{A})} && (3.2)\\ \leq& \kappa\left(D(\mathbf{A})^* D(\mathbf{A})\right)^{1/2} \|R(\mathbf{A})\mathbf{b} - \mathbf{x}_k\|_2 && (3.3) \end{aligned}$$

at each iteration of the algorithm. Recall that $\kappa(\mathbf{A}) = \|\mathbf{A}\|_2\|\mathbf{A}^{-1}\|_2$ is the condition number of $\mathbf{A}$. Further, the condition number of $\mathbf{A}^2$ is approximated by $\kappa(\mathbf{A})\kappa(\mathbf{A})$, and the conditioning of the problem may increase very quickly with the degree of the denominator μ. This can clearly be seen in equation (3.3), where the residual of the modified linear system that GMRES solves, $D(\mathbf{A})\mathbf{x} = N(\mathbf{A})\mathbf{b}$, is related to the residual of the original problem $\mathbf{x} = R(\mathbf{A})\mathbf{b}$ by a factor of $\kappa\left(D(\mathbf{A})^* D(\mathbf{A})\right)^{1/2} = \kappa(D(\mathbf{A}))$.

When computing $R(\mathbf{A})\mathbf{b}$, the two main goals are to have an algorithm that does not require practitioners to modify the preexisting problem and data, and to have an algorithm that has a reliable and efficient way to compute stopping criteria [46]. There are many algorithms satisfying these goals for specific $R(z)$ or $\mathbf{A}$ arising in applications, including when $R(z)$ is a Laplace transform [48], Cauchy-Stieltjes function [64], or ordinary differential equation initial value problem [13, 14], and including when $\mathbf{A}$ is a Kronecker sum [7]. However, there are fewer algorithms for general $R(z)$ and $\mathbf{A}$. Three methods for calculating $R(\mathbf{A})\mathbf{b}$ for general rational functions of a non-Hermitian matrix are GMRES applied to a partial fraction decomposition of the rational function, the Arnoldi method for matrix function approximation (Arnoldi-FA) [21], and the optimal residual Arnoldi method for matrix function approximation (Arnoldi-OR) [23].

GMRES applied to a partial fraction decomposition frequently requires problem modification to transform the rational function $R(z)$ into a sum of rational functions with denominators of degree one, except for repeated roots (see equation (3.6)). This partial fraction

decomposition can be done by hand using some pre-defined formulas, or by using a symbolic toolbox for the implementation language of choice. For example, first define `z` as a symbol, then in MATLAB use `partfrac(R, z, 'FactorMode', 'complex')` or in Python use `sympy.apart(R, full = True).doit()` from the Sympy toolbox. However, the outputs of both of these functions need further modification by the practitioner before they can be used as inputs for the respective built-in GMRES functions. Once $R(\mathbf{A})$ is decomposed into partial fractions, using GMRES requires forming the relation $\mathbf{A}\mathbf{Q}_k = \mathbf{Q}_{k+1}\mathbf{H}_{k+1,k}$, where the columns of $\mathbf{Q}_{k+1}$ form an orthonormal basis of the Krylov subspace $\mathcal{K}_{k+1}(\mathbf{A}, \mathbf{b})$, and using it to optimally minimize the residual for a linear system defined with each partial fraction term. The sum of these residuals can then be used to define a stopping criterion for the algorithm.

Unlike forming a partial fraction decomposition, Arnoldi-FA does not require any problem modification. However, Arnoldi-FA does not have an efficient stopping criterion based on the error or residual since the errors and residuals are inefficient to compute. The iterates of Arnoldi-FA are equal to

$$\mathbf{Q}_k R(\mathbf{H}_k)\mathbf{Q}_k^*\mathbf{b},$$

where the columns of $\mathbf{Q}_k$ form an orthonormal basis of $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$.

Arnoldi-OR is an algorithm that optimizes the residual calculated in the same norm as a basic implementation of GMRES applied to $D(\mathbf{A})\mathbf{x} = N(\mathbf{A})\mathbf{b}$, except where $\mathbf{x}_k$ is in the span of $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ instead of $\mathcal{K}_k(D(\mathbf{A}), \mathbf{b})$. Like Arnoldi-FA, Arnoldi-OR does not require problem modification. Unlike Arnoldi-FA, because Arnoldi-OR minimizes the 2-norm of the residual at each iteration, it has a clear stopping criterion that does not require much extra computation. Arnoldi-OR minimizes the error in the $D(\mathbf{A})^*D(\mathbf{A})$-norm, and it is important to note that if the condition number of $(D(\mathbf{A})^*D(\mathbf{A}))^{1/2}$ is large, then that the 2-norm of the residuals of Arnoldi-OR may differ significantly from the 2-norm of the errors.

In Section 3.1 we derive and define the Arnoldi-OR algorithm, which is our new work [23]. In Section 3.2 we derive *a priori* bounds on the convergence and error for the Arnoldi-

OR algorithm, including bounds for highly non-normal matrices. In Section 3.3 we give numerical examples comparing Arnoldi-OR with GMRES, GMRES on a partial fraction decomposition of $R(\mathbf{A})$, and Arnoldi-FA. The rest of this section is used to review GMRES applied to a partial fraction decomposition of $R(\mathbf{A})$ and Arnoldi-FA.

3.0.1 GMRES Applied to a Partial Fraction Decomposition

Every rational function can be decomposed into a partial fraction decomposition. We start with the simplest case, where the denominator has simple roots, and build up. Let ρ_i be the ith root of $D(z)$ and let all of the roots of $D(z)$ be simple. Let $\mu > \nu$. Recall that $D'(\rho_i) := \lim_{z\to\rho_i} D(z)/(z-\rho_i)$. Then a partial fraction decomposition of $R(z)$ can be written as

$$\frac{N(z)}{D(z)} = \sum_{i=1}^{\mu} \frac{N(\rho_i)/D'(\rho_i)}{z-\rho_i}. \tag{3.4}$$

If $\nu \geq \mu$, then we must first convert $R(z)$ into a mixed expression such that the numerator of the fractional term has degree strictly less than μ. Let $R(z) = P_1(z) + P_2(z)/D(z)$, where $\{P\}_{1,2}$ are polynomials and the maximum degree of $P_2(z)$ is strictly less than μ. Then, assuming all of the roots of $D(z)$ are simple, a partial fraction decomposition can be written as

$$\frac{N(z)}{D(z)} = P_1(z) + \sum_{i=1}^{\mu} \frac{P_2(\rho_i)/D'(\rho_i)}{z-\rho_i}. \tag{3.5}$$

We can directly compute $P_1(\mathbf{A})\mathbf{b}$, so for the rest of this section we assume that $\nu < \mu$ without any loss of generality.

When the roots of $D(z)$ are not simple, a partial fraction decomposition contains summands with polynomials of degree greater than one in the denominator. Let k_i denote the number of times the root ρ_i is repeated, where $\sum_i k_i = \mu$. Then a more general way to define a partial fraction decomposition of $R(Z)$ is

$$\frac{N(z)}{D(z)} = \sum_i \sum_{j=1}^{k_i} \frac{1}{(j-1)!} \frac{d^{j-1}}{d\zeta^{j-1}} \left[\frac{(\zeta-\rho_i)^{k_i} N(\zeta)}{D(\zeta)}\right]_{\zeta\to\rho_i} \frac{1}{(z-\rho_i)^{k_i+1-j}}. \tag{3.6}$$

Note that when $k_i = 1$, the term for the simple root in equation (3.6) is the same as in equation (3.4).

When applying GMRES to a partial fraction decomposition of $R(\mathbf{A})$, GMRES is used to solve the μ linear systems

$$(\mathbf{A} - \rho_i \mathbf{I})^{k_i+1-j} \mathbf{x}_{i_j} = \frac{1}{(j-1)!} \frac{d^{j-1}}{d\zeta^{j-1}} \left[\frac{(\zeta - \rho_i)^{k_i} N(\zeta)}{D(\zeta)} \right]_{\zeta \to \rho_i} \mathbf{b}.$$

The span of the vectors formed by $\mathbf{A} - \rho_i \mathbf{I}$ and $\mathbf{b}$ are the same as the span of the vectors formed by $\mathbf{A}$ and $\mathbf{b}$, $\text{span}\{\mathbf{b}, \mathbf{A}\mathbf{b}, \cdots, \mathbf{A}^{k-1}\mathbf{b}\}$, because $\mathbf{A} - \rho_i \mathbf{I}$ is a linear function of $\mathbf{A}$. This fact means that we only need to form a single orthogonal basis for $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$.

If there are any repeated roots, then we can either use GMRES applied to $(\mathbf{A} - \rho_i \mathbf{I})^{k_i - j + 1}$ or solve the terms with degrees greater than one in the denominator by restarting GMRES with the solution for the same rooted denominator with one fewer degree. For example, let $R(z) = \alpha/(z-2)^3$. Then

$$R(z) = \frac{\alpha_1}{z-2} + \frac{\alpha_2}{(z-2)^2} + \frac{\alpha_3}{(z-2)^3}.$$

Assuming that for all j, $\alpha_j \neq 0$, we would first use GMRES to solve $\frac{1}{\alpha_1}(\mathbf{A} - 2\mathbf{I})\mathbf{x}^{(1)}$ for $\mathbf{x}^{(1)} = \alpha_1(\mathbf{A} - 2\mathbf{I})^{-1}\mathbf{b}$. Then we would solve for $\mathbf{x}^{(2)} = \frac{\alpha_2}{\alpha_1}(\mathbf{A} - 2\mathbf{I})^{-1}\mathbf{x}^{(1)}$. Then finally we would solve for $\mathbf{x}^{(3)} = \frac{\alpha_3 \alpha_1}{\alpha_2}(\mathbf{A} - 2\mathbf{I})^{-1}\mathbf{x}^{(2)}$, and combine these terms to get the solution $\mathbf{x} = R(\mathbf{A})\mathbf{b} = \mathbf{x}^{(1)} + \mathbf{x}^{(2)} + \mathbf{x}^{(3)}$. While using this restarted GMRES implementation we cannot solve for the $\mathbf{x}^{(j)}$ in parallel like we can when computing the solution of simple root terms. Further, in finite arithmetic we would have accumulating error each time we restart GMRES.

One thing to be cautious of when applying GMRES to a partial fraction decomposition is that different terms in a partial fraction decomposition may need more iterations of GMRES to converge to within a desired tolerance. In other words, one cannot simply run GMRES on one of the partial fraction terms until that term converges and be certain that the

Algorithm 8 Arnoldi-FA

1: **Inputs:** $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\mathbf{b} \in \mathbb{C}^n$, N the function handle for the polynomial defining the numerator of $R(z)$, D the function handle for the polynomial defining the denominator of $R(z)$, and k the number of iterations
2: Run Arnoldi on $\mathbf{A}$ and $\mathbf{b}$ for k iterations to get $\mathbf{Q}_k$ and $\mathbf{H}_k$.
3: **Return:** $\mathbf{x}_k = \mathbf{Q}_k\big(D(\mathbf{H}_k)\backslash(N(\mathbf{H}_k)\,(\mathbf{Q}_k^*\mathbf{b}))\big)$

necessary basis has been formed for the rest of the partial fraction terms to also converge. For example, the convergence of terms where ρ_i is in the center of the eigenvalues of $\mathbf{A}$, or if the numerical range of $\alpha_j(\mathbf{A} - 2\mathbf{I})$ contains zero, may be worse than other terms.

3.0.2 Arnoldi-FA

The Arnoldi method for matrix function approximation (Arnoldi-FA) is a general purpose algorithm for calculating the action of a matrix function on a vector. The first step of Arnoldi-FA is to run the Arnoldi algorithm on $\mathbf{A} \in \mathbb{C}^{n\times n}$ for $k+1$ iterations to find the matrix $\mathbf{Q}_k \in \mathbb{C}^{n\times k}$ of orthonormal basis vectors for $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$, and the square upper Hessenberg matrix $\mathbf{H}_k \in \mathbb{C}^{k\times k}$ of coefficients such that

$$\mathbf{A}\mathbf{Q}_k = \mathbf{Q}_k\mathbf{H}_k + \mathbf{H}(k+1,k)\mathbf{q}_{k+1}\mathbf{e}_k^* = \mathbf{Q}_{k+1}\mathbf{H}_{k+1,k}.$$

The vector $\mathbf{e}_k$ is the kth standard basis vector with all zeros except for a one in entry k. The columns of matrix $\mathbf{Q}_{k+1} = [\mathbf{Q}_k\ \mathbf{q}_{k+1}] \in \mathbb{C}^{n\times k+1}$ form an orthonormal basis of $\mathcal{K}_{k+1}(\mathbf{A}, \mathbf{b})$. And the matrix $\mathbf{H}_{k+1,k} \in \mathbb{C}^{k+1\times k}$ is equivalent to $\mathbf{H}_k$ for the first k rows, and the $k+1$st row has zeros in the $k-1$ leftmost entries with the last entry equal to $\mathbf{H}(k+1,k)$.

Then, Arnoldi-FA approximates $R(\mathbf{A})\mathbf{b}$ with $\mathbf{Q}_k R(\mathbf{H}_k)\mathbf{Q}_k^*\mathbf{b}$. For $k < n$,in both exact and finite precision arithmetic, $\mathbf{Q}_k R(\mathbf{H}_k)\mathbf{Q}_k^*\mathbf{b}$ is an approximation because it ignores the error term $\mathbf{H}(k+1,k)\mathbf{q}_{k+1}\mathbf{e}_k^*$. When $k = n$ the error term is zero, so in exact arithmetic the solution is computed. The pseudocode for Arnoldi-FA algorithm is found in Algorithm 8.

Notice that there is no stopping criterion for Arnoldi-FA in Algorithm 8, instead the input k determines the number of iterations run. If the error can be calculated or approximated with a residual at each iteration, then like in Algorithms from Chapter 2, a while loop can be implemented until a desired tolerance for the problem is found. However, calculating the residual of Arnoldi-FA is inefficient.

Using the norm of the residual as the stopping criterion for Arnoldi-FA would require computing

$$\|\mathbf{r}_k\|_2 = \|N(\mathbf{A})\mathbf{b} - D(\mathbf{A})\mathbf{x}_k\|_2$$

at each iteration. The implementation of Arnoldi-FA in Algorithm 8 does not calculate $N(\mathbf{A})\mathbf{b}$ or the action of $D(\mathbf{A})$ on a vector. Instead, Algorithm 8 uses the upper Hessenberg matrix $\mathbf{H}_k$, output from the Arnoldi algorithm, and computes $N(\mathbf{H}_k)$ and $D(\mathbf{H}_k)$. For a dense matrix $\mathbf{A}$, calculating the 2-norm of the residual for Arnoldi-FA at each iteration requires up to an additional $\mu n^2 + 3n$ operations, assuming that $N(\mathbf{A})\mathbf{b}$ is stored in memory. Fewer operations may be required for a sparse matrix.

To better understand the convergence of Arnoldi-FA, we also need to understand its error. Assume that we do not know the true solution $\mathbf{x} = R(\mathbf{A})\mathbf{b}$. Let $p(z)$ be a polynomial with degree less than or equal to $k-1$. Then, the error for Arnoldi-FA is

$$\begin{aligned}
\|R(\mathbf{A})\mathbf{b} - \mathbf{Q}_k R(\mathbf{H}_k)\mathbf{Q}_k^*\mathbf{b}\|_2 \leq & \|R(\mathbf{A})\mathbf{b} - p(\mathbf{A})\mathbf{b}\|_2 + \|p(\mathbf{A})\mathbf{b} - \mathbf{Q}_k p(\mathbf{H}_k)\mathbf{Q}_k^*\mathbf{b}\|_2 \\
& + \|\mathbf{Q}_k p(\mathbf{H}_k)\mathbf{Q}_k^*\mathbf{b} - \mathbf{Q}_k R(\mathbf{H}_k)\mathbf{Q}_k^*\mathbf{b}\|_2 \\
\leq & \|(R(\mathbf{A}) - p(\mathbf{A}))\mathbf{b}\|_2 + 0 \\
& + \|\mathbf{Q}_k \left(p(\mathbf{H}_k) - R(\mathbf{H}_k)\right)\mathbf{Q}_k^*\mathbf{b}\|_2 \\
\leq & \|R(\mathbf{A}) - p(\mathbf{A})\|_2\|\mathbf{b}\|_2 + \|p(\mathbf{H}_k) - R(\mathbf{H}_k)\|_2\|\mathbf{b}\|_2.
\end{aligned}$$

The error of Arnoldi-FA is bounded by how well a polynomial can approximate the rational function of $\mathbf{A}$, and how well that same polynomial can approximate the rational function of $\mathbf{H}_k$. However, $\mathbf{H}_k$ depends on both $\mathbf{A}$ and $\mathbf{b}$ and is not known before running the

algorithm. Further, how well a rational function is approximated by a polynomial depends on the location of the rational function's poles and the proximity of those poles to the eigenvalues of the matrix input to both functions. However, at each iteration of Arnoldi, the eigenvalues of $\mathbf{H}_k$ are likely to change. Some of these eigenvalues of $\mathbf{H}_k$ will likely converge to eigenvalues of $\mathbf{A}$, and in practice usually the extreme eigenvalues (that have the greatest absolute value) of $\mathbf{A}$ are converged to first. However, before an eigenvalue of $\mathbf{H}_k$ convergences to an eigenvalue of $\mathbf{A}$, the eigenvalues of $\mathbf{H}_k$ may be located anywhere within the numerical range of $\mathbf{A}$, $W(\mathbf{A})$. This means that it can be impossible to predict how well a polynomial approximates a rational function on the kth iteration, especially if $R(z)$ has a pole within $W(\mathbf{A})$, without first calculating the upper Hessenberg matrix for that iteration of Arnoldi.

Not only is calculating a stopping criterion for Arnoldi-FA a significant amount of extra work at each iteration, Arnoldi-FA does not converge monotonically, which means that it is possible for $\|\mathbf{r}_k\|_2 > \|\mathbf{r}_{k-1}\|_2$. In Figures 3.1 and 3.5 from Section 3.3 are examples where the convergence of Arnoldi-FA has significant spikes in the 2-norms of the residuals and the errors. Regardless, Arnoldi-FA is a relatively efficient method for computing the action of a general matrix function on a vector when the matrix is non-Hermitian.

3.1 Arnoldi-OR

The optimal residual Arnoldi method for matrix function approximation (Arnoldi-OR) is an algorithm to compute $\mathbf{x} = R(\mathbf{A})\mathbf{b}$. Let $R(\mathbf{A}) =$
$D(\mathbf{A})^{-1}N(\mathbf{A})$, then the residual of Arnoldi-OR is optimized in the 2-norm, which means that the error $r(\mathbf{A})\mathbf{b} - \mathbf{x}_k$ is optimized in the $D(\mathbf{A})^*D(\mathbf{A})$-norm. This norm in which Arnoldi-OR optimizes the error is the same as the norm in which GMRES optimizes the error when applied to $D(\mathbf{A})\mathbf{x} = N(\mathbf{A})\mathbf{b}$. However, unlike GMRES applied to $D(\mathbf{A})\mathbf{x} = N(\mathbf{A})\mathbf{b}$ which finds $\mathbf{x}_k$ in $\mathcal{K}_k(D(\mathbf{A}), \mathbf{b})$, Arnoldi-OR finds an $\mathbf{x}_k$ in $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$. The solution $\mathbf{x}$ can often be more efficiently approximated using $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$.

We now derive the Arnoldi-OR algorithm. We assume that $\|\mathbf{b}\|_2 = 1$ without any loss

of generality, which allows us to define $\mathbf{q}_1 = \mathbf{b}$. We also assume that $\mathbf{x}_0 = \mathbf{0}$ to simplify the presentation, but the following holds for general $\mathbf{x}_0$. For the system $D(\mathbf{A})\mathbf{x} = N(\mathbf{A})\mathbf{b}$, GMRES finds a vector $\mathbf{y}$ such that the 2-norm of the residual is minimized. Then the residual is

$$\begin{aligned}
\|N(\mathbf{A})\mathbf{b} - D(\mathbf{A})\mathbf{x}_k\|_2^2 =& \|N(\mathbf{A})\mathbf{b} - D(\mathbf{A})\mathbf{Q}_k\mathbf{y}\|_2^2 \\
=& \|D(\mathbf{A})[D(\mathbf{A})^{-1}N(\mathbf{A})\mathbf{b} - \mathbf{Q}_k\mathbf{y}]\|_2^2 \\
=& \|D(\mathbf{A})[R(\mathbf{A})\mathbf{b} - \mathbf{Q}_k\mathbf{y}]\|_2^2 \\
=& \langle R(\mathbf{A})\mathbf{b} - \mathbf{Q}_k\mathbf{y}, [D(\mathbf{A})^* D(\mathbf{A})][R(\mathbf{A})\mathbf{b} - \mathbf{Q}_k\mathbf{y}]\rangle \\
=& \|R(\mathbf{A})\mathbf{b} - \mathbf{Q}_k\mathbf{y}\|_{D(\mathbf{A})^* D(\mathbf{A})}^2.
\end{aligned}$$

However, on this system GMRES forms a basis for the Krylov subspace $\mathcal{K}(D(\mathbf{A}), \mathbf{b}) = \text{span}\{\mathbf{b}, D(\mathbf{A})\mathbf{b}, D^2(\mathbf{A})\mathbf{b}, \cdots\}$. Anytime μ, the maximum degree of $D(z)$, is greater than one, then we are skipping powers of $\mathbf{A}$ while forming the Krylov subspace. Since we have access to $\mathbf{A}$, we want to take advantage of each matrix-vector multiplication performed in the Arnoldi algorithm and form a Krylov subspace for $\mathcal{K}(\mathbf{A}, \mathbf{b}) = \text{span}\{\mathbf{b}, \mathbf{A}\mathbf{b}, \mathbf{A}^2\mathbf{b}, \cdots\}$ instead. We can use the relationship $\mathbf{A}\mathbf{Q}_k = \mathbf{Q}_{k+1}\mathbf{H}_{k+1,k}$ to rewrite $\mathbf{A}$ and its powers as:

$$\begin{aligned}
\mathbf{A}\mathbf{Q}_k =& \mathbf{Q}_{k+1}\mathbf{H}_{k+1,k} \\
\mathbf{A}^2\mathbf{Q}_k =& \mathbf{A}\mathbf{Q}_{k+1}\mathbf{H}_{k+1,k} = \mathbf{Q}_{k+2}\mathbf{H}_{k+2,k+1}\mathbf{H}_{k+1,k} := \mathbf{Q}_{k+2}\mathbf{H}_{k+2,k} \\
&\vdots \\
\mathbf{A}^m\mathbf{Q}_k =& \mathbf{Q}_{k+m}\mathbf{H}_{k+m,k+m-1}\cdots\mathbf{H}_{k+1,k} := \mathbf{Q}_{k+m}\mathbf{H}_{k+m,k}.
\end{aligned}$$

This relation for $\mathbf{A}^m\mathbf{Q}_k$ assumes that we can run Arnoldi m additional iterations, and as long as $k + m \leq n$, this assumption is true. When $k + m \geq n$ then Arnoldi-OR finds the true solution.

Define $N(z) = \sum_{l=0}^{\nu} c_l z^l$ and $D(z) = \sum_{m=0}^{\mu} d_m z^m$. To simplify the summations, we

define $\mathbf{H}_{k,k} := \mathbf{I}_k$, and note that $\mathbf{H}_k \neq \mathbf{H}_{k,k} = \mathbf{I}_k$. Using the assumptions $\|\mathbf{b}\|_2 = 1$, $\mathbf{q}_1 = \mathbf{b}$, and $\mathbf{x}_0 = \mathbf{0}$, then the residual $\|N(\mathbf{A})\mathbf{b} - D(\mathbf{A})\mathbf{x}_k\|_2$ is equivalent to $\min_{\mathbf{y}} \|N(\mathbf{A})\mathbf{Q}_k\mathbf{e}_1 - D(\mathbf{A})\mathbf{Q}_k\mathbf{y}\|_2$. By plugging in the summation forms of $D(z)$ and $N(z)$ into the residual, we can write

$$\min_{\mathbf{y}} \left\| \sum_{l=0}^{\nu} c_l \mathbf{A}^l \mathbf{Q}_k \mathbf{e}_1 - \sum_{m=0}^{\mu} d_m \mathbf{A}^m \mathbf{Q}_k \mathbf{y} \right\|_2 = \min_{\mathbf{y}} \left\| \sum_{l=0}^{\nu} c_l \mathbf{Q}_{k+l} \mathbf{H}_{k+l,k} \mathbf{e}_1 - \sum_{m=0}^{\mu} d_m \mathbf{Q}_{k+m} \mathbf{H}_{k+m,k} \mathbf{y} \right\|_2 . \tag{3.7}$$

Let $\tau := \max\{\nu, \mu\}$ be the maximum degree of the numerator and denominator polynomials that make up $R(z)$. Let $\hat{\mathbf{H}}_{k+m,k} \in \mathbb{C}^{k+\tau \times k}$ be the coefficient matrix $\mathbf{H}_{k+m,k}$ with $\tau - m$ rows of zeros appended to the bottom. Then the residual in equation (3.7) can be rewritten as

$$\min_{\mathbf{y}} \left\| \sum_{l=0}^{\nu} c_l \mathbf{Q}_{k+\tau} \hat{\mathbf{H}}_{k+l,k} \mathbf{e}_1 - \sum_{m=0}^{\mu} d_m \mathbf{Q}_{k+\tau} \hat{\mathbf{H}}_{k+m,k} \mathbf{y} \right\|_2 = \min_{\mathbf{y}} \left\| \sum_{l=0}^{\nu} c_l \hat{\mathbf{H}}_{k+l,k} \mathbf{e}_1 - \sum_{m=0}^{\mu} d_m \hat{\mathbf{H}}_{k+m,k} \mathbf{y} \right\|_2 . \tag{3.8}$$

This minimization problem can be solved for example using QR factorization of the coefficient matrix, which can also be computed and updated iteratively through the use of Givens rotations.

Let $\mathbf{H}$ be the square coefficient matrix output by Arnoldi on the nth iteration, then the submatrices of $\mathbf{H}$ calculated during successive iterations of Arnoldi satisfy

$$\mathbf{H}_{k+2,k+1} = \begin{bmatrix} \mathbf{H}_{k+1,k} & \mathbf{H}(1:k+1, k+1) \\ \mathbf{0}_k^T & \mathbf{H}(k+2, k+1) \end{bmatrix} .$$

Each time an upper Hessenberg matrix is multiplied by itself, the resulting matrix contains another non-zero subdiagonal. Let $\mathbf{J}$ be any upper Hessenberg matrix. The matrix $\mathbf{J}$

is guaranteed to have only zeros below the first sub-diagonal. The matrix $\mathbf{J}^2$ is guaranteed to have only zeros below the second sub-diagonal. Likewise, the matrix $\mathbf{J}^m$ is guaranteed to have only zero entries below the mth sub-diagonal. For example, if $\mathbf{J}$ is a 6×6 Upper Hessenberg matrix, then

$$\mathbf{J} = \begin{bmatrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & 0 & * & * & * & * \\ 0 & 0 & 0 & * & * & * \\ 0 & 0 & 0 & 0 & * & * \end{bmatrix}, \quad \mathbf{J}^2 = \begin{bmatrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \\ 0 & * & * & * & * & * \\ 0 & 0 & * & * & * & * \\ 0 & 0 & 0 & * & * & * \end{bmatrix}, \text{ and}$$

$$\mathbf{J}^4 = \begin{bmatrix} * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \\ * & * & * & * & * & * \\ 0 & * & * & * & * & * \end{bmatrix}.$$

In the example above, we can see that in columns 1-3 of $\mathbf{J}^2$, all the entries below row 5 are zero. Since $\mathbf{H}_{k+2,k+1}$ and $\mathbf{H}_{k+1,k}$ are submatrices of $\mathbf{H}$, then $\mathbf{H}_{k+2,k+1}\mathbf{H}_{k+1,k}$ is equal to the first $k+2$ rows and k columns of $\mathbf{H}^2$.

One way that the Arnoldi-OR algorithm takes advantage of this relationship for upper Hessenberg matrices is by using the fact that all of the non-zero entries of $N(\mathbf{H}_{k+\tau})(:,1)$ are calculated during the first iteration. Since $N(z)$ has maximum degree $\nu \leq \tau$, $N(\mathbf{H}_{k+\tau})(k+\tau, 1) = 0$ for $k \geq 2$. We can also rewrite equation (3.8) using this relationship

for Upper Hessenberg matrices, allowing for a more efficient implementation of Arnoldi-OR:

$$\min_{\mathbf{y}} \left\| \sum_{l=0}^{\nu} c_l \hat{\mathbf{H}}_{k+l,k} \mathbf{e}_1 - \sum_{m=0}^{\mu} d_m \hat{\mathbf{H}}_{k+m,k} \mathbf{y} \right\| = \min_{\mathbf{y}} \left\| N(\mathbf{H}_{k+\tau})(:,1) - D(\mathbf{H}_{k+\tau})(:,1:k)\mathbf{y} \right\|_2 . \tag{3.9}$$

We are now ready to define the basic version of Arnoldi-OR in Algorithm 10. This basic

Algorithm 9 Basic Arnoldi-OR

1: **Inputs:** $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\mathbf{b} \in \mathbb{C}^n$, N the function handle for the polynomial defining the numerator of $R(\cdot)$, ν the max degree of $N(\cdot)$, D the function handle for the polynomial defining the denominator of $R(\cdot)$, μ the max degree of $D(\cdot)$, and kmax the total number of iterations, where kmax$+\max\{\nu,\mu\} < n$
2: $\tau = \max\{\nu,\mu\}$
3: Run Arnoldi for kmax $+\,\tau$ iterations to get the orthogonal basis matrix $\mathbf{Q}_{\text{kmax}+\tau}$ and square upper Hessenberg coefficient matrix $\mathbf{H}_{\text{kmax}+\tau}$.
4: Calculate and store $\mathbf{J} := D(\mathbf{H}_{\text{kmax}+\tau})(1:\text{kmax}+\tau, 1:\text{kmax})$.
5: Calculate and store $\mathbf{n} = N(\mathbf{H}_{\text{kmax}+\tau})(1:\text{kmax}+\tau, 1)$.
6: Solve $\min_{\mathbf{y}} \|\mathbf{n} - \mathbf{J}\mathbf{y}\|_2$.
7: $\|\mathbf{r}_{\text{kmax}}\|_2 = \|\mathbf{n} - \mathbf{J}\mathbf{y}\|_2$, where $\mathbf{r}_{\text{kmax}}$ is the residual vector for the kmax iteration.
8: **Return:** $\mathbf{x}_{\text{kmax}} = \mathbf{Q}_{\text{kmax}}\mathbf{y}$.

implementation of Arnoldi-OR has similar shortcomings to the basic GMRES in Chapter 2, Algorithm 4. If we efficiently solve for the residual at each iteration, then we can include a stopping criterion based on reaching a desired tolerance before kmax iterations are computed. However, computing line 6 in Algorithm 9 can easily be inefficient, for example by using QR factorization on $\mathbf{J}$ at each iteration. One alternative to calculating the residual at each iteration could be to use error bounds to determine the number of iterations, kmax, to input to Algorithm 9 that would achieve at least a desired tolerance. Error bounds for the Arnoldi-OR algorithm are found in Section 3.2. However, these error bounds tend to overestimate the error, so using them would likely lead to extra iterations of Arnoldi-OR being computed. Instead of using error bounds to bound kmax, we can use Givens rotations to efficiently minimize $\mathbf{y}$ and check the residual against the tolerance at each iteration,

similar to Algorithm 5 from Chapter 2. Algorithm 10 on the next page incorporates Givens rotations into the pseudocode for Arnoldi-OR.

Algorithm 10 Arnoldi-OR with Givens Rotations

Inputs: $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\mathbf{b} \in \mathbb{C}^n$, N the function handle for the polynomial defining the numerator of $R(\cdot)$, ν the max degree of $N(\cdot)$, D the function handle for the polynomial defining the denominator of $R(\cdot)$, μ the max degree of $D(\cdot)$, and the tolerance ϵ
$\tau = \max\{\mu, \nu\}$
Run $1+\tau$ iterations of Arnoldi to get the orthogonal basis matrix $\mathbf{Q}_{1+\tau}$ and square upper Hessenberg coefficient matrix $\mathbf{H}_{1+\tau}$
$\mathbf{J} := D(\mathbf{H}_{1+\tau})(1 : 1+\tau, 1)$
$\mathbf{n} = N(\mathbf{H}_{1+\tau})(1 : 1+\tau, 1)$
Form the Givens rotations matrices $\mathbf{G}_{1,2}$, $\mathbf{G}_{1,3}, \cdots$, $\mathbf{G}_{1,1+\mu}$ to annihilate the entries $\mathbf{J}(2 : 1+\mu)$
$\mathbf{U} = \mathbf{G}_{1,2} \cdots \mathbf{G}_{1,1+\mu}\mathbf{J}$
$\mathbf{n} = \mathbf{G}_{1,2} \cdots \mathbf{G}_{1,1+\mu}\mathbf{n}$
Solve $\mathbf{U}(1,1)\mathbf{y}_1 = \mathbf{n}(1)$
Set $k = 1$ and the residual norm $r = \|\mathbf{n} - \mathbf{U}\mathbf{y}\|_2$.
while $r > \epsilon$ **do**
    $k \leftarrow k + 1$
    Run the next iteration of Arnoldi to get $\mathbf{Q}_{k+\tau}$ and $\mathbf{H}_{k+\tau}$
    Append a row of zeros to the bottoms of $\mathbf{U}$ and $\mathbf{n}$
    Form the kth column of $D(\mathbf{H}_{k+\tau})$
    Apply the previous Givens rotations to $D(\mathbf{H}_{k+\tau})(1 : k+\tau, k)$; $\mathbf{U}(1 : k+\tau, k) = (\mathbf{G}_{k-1,2} \cdots \mathbf{G}_{k-1,1+\mu}) \cdots (\mathbf{G}_{1,2} \cdots \mathbf{G}_{1,1+\mu})D(\mathbf{H}_{k+\tau})(1 : k+\tau, k)$
    Form $\mathbf{G}_{k,k+1}, \cdots, \mathbf{G}_{k,k+\mu}$ to annihilate the entries in rows $k+1$ through $k+\mu$ in column k of $\mathbf{U}$
    $\mathbf{U}(1 : k+\tau, k) \leftarrow \mathbf{G}_{k,k+1} \cdots \mathbf{G}_{k,k+\mu}\mathbf{U}(1 : k+\tau, k)$
    $\mathbf{n} \leftarrow \mathbf{G}_{k,k+1} \cdots \mathbf{G}_{k,k+\mu}\mathbf{n}$
    Solve the upper triangular linear system $\mathbf{U}(1 : k, 1 : k)\mathbf{y} = \mathbf{n}(1 : k)$, for example using back substitution
    $r \leftarrow \|\mathbf{n} - \mathbf{U}\mathbf{y}\|_2$
end while
Return: $\mathbf{x}_k = \mathbf{Q}_k\mathbf{y}_k$

A MATLAB implementation of Arnoldi-OR can be found at `https://www.github.com/tygris/ArnoldiOR`.

3.2 Error Bounds for Arnoldi-OR

One strength of the Arnoldi-OR algorithm is that we can bound the method's error *a priori*. Let $\mathbf{S} := D(\mathbf{A})^* D(\mathbf{A})$. The Arnoldi-OR approximation, given by $\mathbf{Q}_k\mathbf{y}$, is equivalent to $\mathbf{P}_{k-1}(\mathbf{A})\mathbf{b}$, where $\mathbf{P}_{k-1}$ is the $(k-1)$-degree polynomial that minimizes the $\mathbf{S}$-norm of the error. This relationship leads to the bound:

$$\frac{\|R(\mathbf{A})\mathbf{b} - \mathbf{Q}_k\mathbf{y}\|_2}{\|\mathbf{b}\|_2} \leq \kappa(\mathbf{S})^{1/2} \min_{p_{k-1}\in\mathcal{P}_{k-1}} \|R(\mathbf{A}) - p_{k-1}(\mathbf{A})\|_2, \tag{3.10}$$

where $\kappa(\mathbf{S}) := \|\mathbf{S}\|_2\|\mathbf{S}^{-1}\|_2$ is the condition number of $\mathbf{S}$.

The bound (3.10) is difficult to work with since we are comparing the rational function of a matrix to polynomials of a matrix. As we discussed earlier in Section 3.0.2, how well a specific polynomial of a matrix $\mathbf{A}$ approximates a rational function of $\mathbf{A}$ will be affected by the proximity of the poles in the rational function to the eigenvalues of $\mathbf{A}$, but not entirely determined by that proximity.

If $\mathbf{A}$ is diagonalizable with eigendecomposition $\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1}$, where $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$, the bound refines to:

$$\frac{\|R(\mathbf{A})\mathbf{b} - \mathbf{Q}_k\mathbf{y}\|_2}{\|\mathbf{b}\|_2} \leq \kappa(\mathbf{S})^{1/2}\kappa(\mathbf{V}) \min_{p_{k-1}\in\mathcal{P}_{k-1}} \max_{i=1,\ldots,n} |R(\lambda_i) - p_{k-1}(\lambda_i)|. \tag{3.11}$$

One reason to prefer bound (3.11) to (3.10) is that now we are comparing how well the rational function is approximated by a polynomial on a finite set of points in the complex plane. However, in exchange for using a smaller search space in the optimization, we now have a constant factor of $\kappa(\mathbf{V})$ out front of the bound. As we discussed in Chapter 2 in relation to GMRES, the error bound based on the eigenvalues of $\mathbf{A}$ works well as long as the matrix $\mathbf{A}$ has near perpendicular eigenvectors. However, we are also interested in the case

where $\mathbf{A}$ is highly non-normal, which means that $\kappa(\mathbf{V})$ is large because the eigenvectors are not nearly perpendicular. In fact, $\mathbf{A}$ may have one or more pairs of eigenvectors that are nearly parallel.

For any monotonic convergence curve, one may choose any set of eigenvalues and find a matrix $\mathbf{A}$ with those eigenvalues and a right hand side vector $\mathbf{b}$ such that when Arnoldi-OR is applied to $\mathbf{A}$ and $\mathbf{b}$ the resulting convergence curve will follow the monotonic convergence curve chosen first. This phenomenon was first proven for GMRES, which is the special case of Arnoldi-OR where the denominator polynomial has max degree one [59]. To see how this research extends to the Arnoldi-OR algorithm for more rational functions, the reader is directed to Section 5 of [23].

Like with the GMRES algorithm, K-spectral set theory can be applied to improve the error bound for highly non-normal matrices. For non-diagonalizable $\mathbf{A}$ or for $\mathbf{A}$ with large $\kappa(\mathbf{V})$, the error can also be related to how well $R(z)$ can be approximated over the numerical range, $W(\mathbf{A}) := \{\langle \mathbf{q}, \mathbf{Aq}\rangle : \langle \mathbf{q}, \mathbf{q}\rangle = 1\}$. As shown in [30], $W(\mathbf{A})$ is a $(1+\sqrt{2})$-spectral set, leading to the bound:

$$\frac{\|R(\mathbf{A})\mathbf{b} - \mathbf{Q}_k\mathbf{y}\|_2}{\|\mathbf{b}\|_2} \leq \kappa(\mathbf{S})^{1/2}(1+\sqrt{2}) \min_{p_{k-1}\in\mathcal{P}_{k-1}} \max_{z\in W(\mathbf{A})} |R(z) - p_{k-1}(z)|. \tag{3.12}$$

In the bound (3.12), the approximation of a rational function by a polynomial is compared in a region of the complex plane containing the eigenvalues of $\mathbf{A}$ instead of only in a finite set of points like in the bound (3.11). In exchange for comparing the function approximation over a larger set of points, we have replaced $\kappa(\mathbf{V})$ with the smaller constant $1+\sqrt{2}$ for highly non-normal matrices. Similarly, we can remove $\kappa(\mathbf{S})^{1/2}$ if we replace the 2-norm of the error with the $\mathbf{S}$-norm of the error, and replace $W(\mathbf{A})$ with $W_{\mathbf{S}}(\mathbf{A}) := \{\langle \mathbf{q}, \mathbf{SAq}\rangle : \langle \mathbf{q}, \mathbf{Sq}\rangle = 1\} = W(\mathbf{S}^{1/2}\mathbf{AS}^{-1/2})$. Then the error bound that uses

the $\mathbf{S}$-norm and $W_{\mathbf{S}}(\mathbf{A})$ is:

$$\frac{\|R(\mathbf{A})\mathbf{b}-\mathbf{Q}_k\mathbf{y}\|_{\mathbf{S}}}{\|\mathbf{b}\|_{\mathbf{S}}} \leq (1+\sqrt{2}) \min_{p_{k-1}\in\mathcal{P}_{k-1}} \max_{z\in W(\mathbf{S}^{1/2}\mathbf{A}\mathbf{S}^{-1/2})} |R(z)-p_{k-1}(z)|. \tag{3.13}$$

If there is a pole located within either $W(\mathbf{A})$ or $W(\mathbf{S}^{1/2}\mathbf{A}\mathbf{S}^{-1/2})$, then the bounds (3.12) and (3.13) are not useful, respectively, since they would have infinite values on the right hand side.

By removing the poles of $R(z)$ from the sets over which the maximum of $|R(z)-p_{k-1}(z)|$ is calculated, we can write a finite error bound for Arnoldi-OR. Recall that the numerical radius of $\mathbf{A}$ is $w(\mathbf{A}) := \max\{|\langle \mathbf{q}, \mathbf{A}\mathbf{q}\rangle : \langle \mathbf{q},\mathbf{q}\rangle = 1\}$. Then, as shown in [29], we can remove a disk of radius $1/w((\xi\mathbf{I}-\mathbf{A})^{-1})$ centered at the pole of $R(z)$, $\xi \in W(\mathbf{A})$, to obtain a $(3+2\sqrt{3})$-spectral set. The resulting error bound is:

$$\frac{\|R(\mathbf{A})\mathbf{b}-\mathbf{Q}_k\mathbf{y}\|_2}{\|\mathbf{b}\|_2} \leq \kappa(\mathbf{S})^{1/2}(3+2\sqrt{3}) \min_{p_{k-1}\in\mathcal{P}_{k-1}} \max_{z\in W(\mathbf{A})\setminus\mathcal{D}(\xi,1/w((\xi\mathbf{I}-\mathbf{A})^{-1}))} |R(z)-p_{k-1}(z)|. \tag{3.14}$$

Chapter 5 goes through work extending this result to remove m disks from the numerical range, which we apply here to get the error bound:

$$\frac{\|R(\mathbf{A})\mathbf{b}-\mathbf{Q}_k\mathbf{y}\|_2}{\|\mathbf{b}\|_2} \leq \kappa(\mathbf{S})^{1/2}\left(1+2m+\sqrt{(1+2m)^2+2m+1}\right) \min_{p_{k-1}\in\mathcal{P}_{k-1}} \max_{z\in W(\mathbf{A})\setminus\cup_{j=1}^{m}\mathcal{D}(\xi_j,1/w((\xi_j\mathbf{I}-\mathbf{A})^{-1}))} |R(z)-p_{k-1}(z)|. \tag{3.15}$$

Each bound in equations (3.10) through (3.15) relies on more than just the eigenvalues of $\mathbf{A}$. However, all of these bounds are based on information that we can gain from $\mathbf{A}$ directly without needing to run the Arnoldi algorithm first, thus allowing us to bound the

error of Arnoldi-OR *a priori*.

3.3 Numerical Examples

Here we present numerical experiments comparing the 2-norm optimal approximation from the Krylov subspaces $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ (which we compute by directly evaluating $R(\mathbf{A})\mathbf{b}$ and finding its orthogonal projection onto the Krylov subspace) with the four algorithms discussed in this chapter: Arnoldi-OR, Arnoldi-FA, GMRES applied to the linear system $D(\mathbf{A})\mathbf{x} = N(\mathbf{A})\mathbf{b}$, and GMRES applied to a partial fraction decomposition of $R(\mathbf{A})$. We compare the 2-norms of the residuals $\|N(\mathbf{A})\mathbf{b} - D(\mathbf{A})\mathbf{x}_k\|_2$ and the 2-norms of the errors $\|D(\mathbf{A})^{-1}N(\mathbf{A})\mathbf{b} - \mathbf{x}_k\|_2$ for each of these approximations. We run all of the experiments in MATLAB in finite, 16-digit, precision arithmetic. All of the randomly generated values are calculated using the standard normal distribution, and complex values are generated by summing one real and imaginary randomly generated term; for example, to randomly generate μ complex polynomial roots in MATLAB we use `randn(1, mu) + 1i*randn(1, mu)`. All random values are generated using the default seed (`rng('default')`) and initialized in the order presented in this section unless otherwise stated. We also consider various error bounds for these quantities. For a comprehensive comparison of different error bounds for GMRES, see [42].

The code used to generate the figures can be found at: `https://github.com/tygris/arnoldior`.

3.3.1 Random **A**

Let $\mathbf{A}$ be the sum of a 100 by 100 real random matrix, $\mathbf{R}$, and a multiple of the identity. Let $\mathbf{b}$ be a real random vector of length 100 that we normalize for convenience. Let $D(z)$ be a random cubic and $N(z)$ be a random quadratic function defined by:

$$D(z) = \gamma(z - r_1)(z - r_2)(z - r_3), \quad N(z) = \delta(z - s_1)(z - s_2),$$

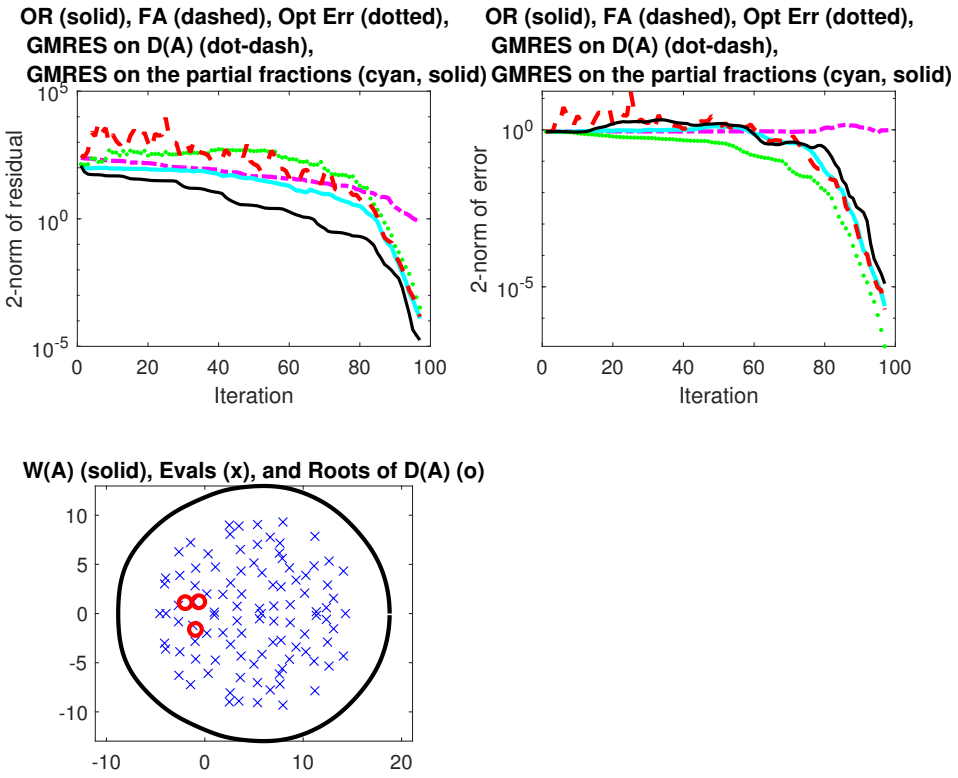


Figure 3.1: $\mathbf{A} = \text{randn}(100) + 5\text{I}$, $\mathbf{b} = \text{randn}(100, 1)$, $N(z)$ random quadratic, $D(z)$ random cubic, generated in MATLAB with the default seed in that order. Top plots show the 2-norms of the residuals and 2-norms of the errors for Arnoldi-OR (black, solid), Arnoldi-FA (red, dashed), GMRES applied to $D(\mathbf{A})$ (magenta, dot-dash), GMRES applied to a partial fraction decomposition (cyan, solid), and the 2-norm optimal Krylov approximation using $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ (green, dotted). Lower left plot shows eigenvalues (blue, 'x's), $\partial W(\mathbf{A})$ (black, solid), and poles of $R(z)$ (red, 'o's).

where $\gamma, r_1, r_2, r_3, \delta, s_1, s_2$ are complex numbers. The roots of $D(z)$ may lie within the numerical range of $\mathbf{A}$ depending on the multiple of the identity added to the randomly generated matrix.

In Figure 3.1, $\mathbf{A} = \mathbf{R} + 5\mathbf{I}$, where $\mathbf{R}$ is the real random 100 by 100 matrix initialized using the standard normal distribution. The condition number of the eigenvector matrix of $\mathbf{A}$, $\kappa(\mathbf{V})$, is approximately 94.75, and $\kappa(D(\mathbf{A})^* D(\mathbf{A}))^{1/2}$ is approximately 2.3e+4. The upper left plot shows the 2-norm of the residuals, $\|N(\mathbf{A})\mathbf{b} - D(\mathbf{A})\mathbf{x}_k\|_2$, for the first 97

iterations for all four methods tested and the 2-norm optimal Krylov approximation. Since $D(z)$ is a cubic function, the maximum number of iterations for Arnoldi-OR is 97, but the other three methods can be run for 100 iterations. Necessarily, the 2-norm of the Arnoldi-OR residual (black, solid) is the smallest at each iteration. The upper right plot shows the 2-norm of the errors, $\|D(\mathbf{A})^{-1}N(\mathbf{A})\mathbf{b} - \mathbf{x}_k\|_2$, at each iteration. Necessarily, the error of the 2-norm optimal Krylov approximation (green, dotted), which is the orthogonal projection of the true solution into $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$, is the smallest at each iteration. At the 97th iteration, Arnoldi-OR had an error of approximately 1e-2, similar to Arnoldi-FA and GMRES applied to a partial fraction decomposition. For both the 2-norms of the residuals and errors in Figure 3.1, we see that Arnoldi-FA, and to a lesser extent GMRES applied to partial fraction decomposition, do not converge monotonically. Finally, the error of GMRES applied to a partial fraction decomposition is consistently one of the best throughout all iterations after the optimal approximation in the Krylov subspace.

The lower left plot of Figure 3.1 shows the boundary of the numerical range of $\mathbf{A}$ (black, solid), the eigenvalues of $\mathbf{A}$ (blue, 'x's), and the roots of $D(\mathbf{A})$ (red, 'o's). While each root is close to an eigenvalue of $\mathbf{A}$, there is no overlap. For all five methods, the convergence is very slow. To bound the error of Arnoldi-OR, we could use bound (3.15) and remove three disks from the numerical range centered at the roots of $D(\mathbf{A})$. However, computing the best uniform polynomial approximation to $R(z)$ on the multiply connected domain consisting of $W(\mathbf{A})$ without three disks centered at each pole of $R(z)$ is difficult. Instead, one could bound the error of Arnoldi-OR using bound (3.11) by computing the best polynomial approximation over a finite set of points. In this example, $\kappa(\mathbf{V}) = 94.75$ is of modest size.

In Figure 3.2, the randomly generated values for $\mathbf{R}$, $\mathbf{b}$, $D(z)$, and $N(z)$ are all equivalent to those used to generate Figure 3.1 and now $\mathbf{A} = \mathbf{R} + 15\mathbf{I}$. The condition number of the eigenvector matrix of $\mathbf{A}$ is the same as before, about 94.75, but now $\kappa(D(\mathbf{A})^* D(\mathbf{A}))^{1/2}$ is reduced to approximately 227.8. The upper plots show that convergence is much faster than in Figure 3.1, and that Arnoldi-OR, Arnoldi-FA, GMRES on partial fractions, and the

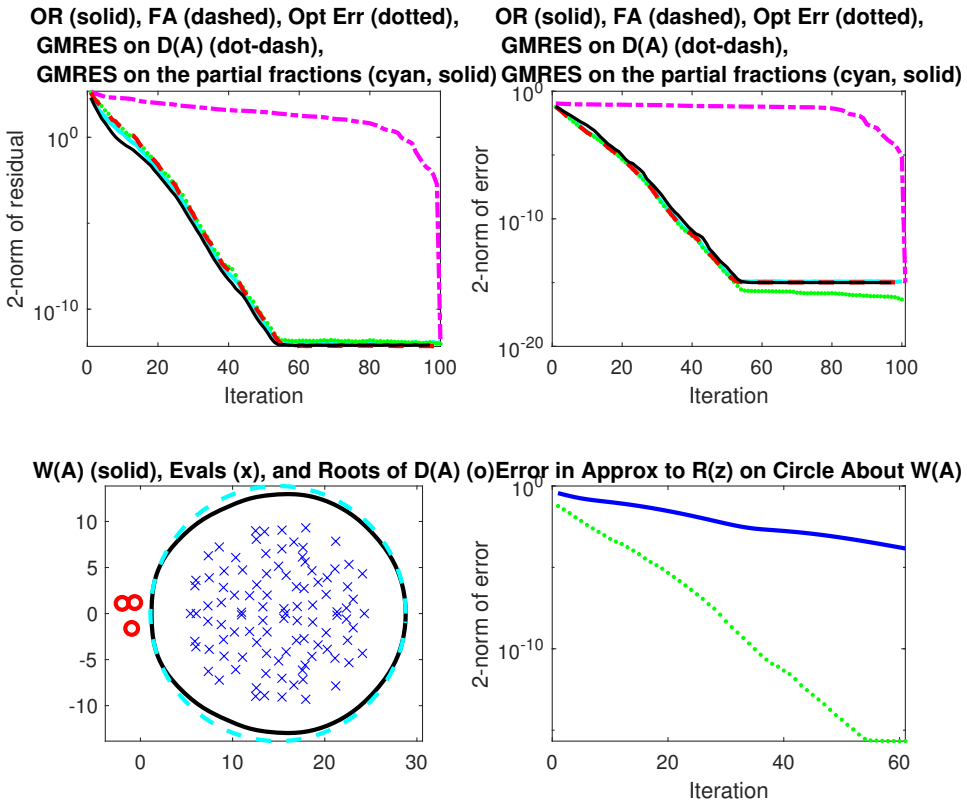


Figure 3.2: $\mathbf{A} = \text{randn}(100) + 15\mathbf{I}$, $\mathbf{b} = \text{randn}(100, 1)$, $N(z)$ random quadratic, and $D(z)$ random cubic, were generated in MATLAB with the seed rng('default') in the order presented, and are equivalent to the randomly generated values used in Figure 3.1. Top plots show the 2-norms of the residuals and the 2-norms of the errors from Arnoldi-OR (black, solid), Arnoldi-FA (red, dashed), GMRES applied to $D(\mathbf{A})$ (magenta, dot-dash), GMRES applied to a partial fraction decomposition (cyan, solid), and the 2-norm optimal approximation from $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ (green, dotted). Lower left plot shows eigenvalues of $\mathbf{A}$ (blue, 'x's), $\partial W(\mathbf{A})$ (black, solid), and poles of $R(z)$ (red, 'o's). Lower right plot shows a comparison between the error of the 2-norm optimal Krylov approximation (green, dotted) and a numerical approximation of $\min_{p_{k-1} \in \mathcal{P}_{k-1}} \sup_{z \in \mathcal{D}(14.9,13.9)} |R(z) - p_{k-1}(z)|$ (blue, solid) which can be used in the error bound (3.12).

2-norm optimal Krylov approximation converge in about 60 iterations. However, GMRES applied to $D(\mathbf{A})$ takes the full 100 iterations to converge, despite optimizing the 2-norm of the residual and the $D(\mathbf{A})^* D(\mathbf{A})$-norm of the error, the same norms as Arnoldi-OR.

The lower left plot shows the boundary of the numerical range of $\mathbf{A}$ (black, solid), the

eigenvalues of $\mathbf{A}$ (blue, 'x's), and the roots of $D(\mathbf{A})$ (red, 'o's). In Figure 3.2 the roots of $D(\mathbf{A})$ are not close to any eigenvalues and all lie outside the boundary of the numerical range of $\mathbf{A}$. Since there are no roots within the numerical range, we can use (3.12) to bound the error, and this is useful because $1 + \sqrt{2} \ll \kappa(V) \approx 94.75$. However, finding the best polynomial approximation of $R(z)$ over the numerical range, even without any disks removed, is difficult. Since the numerical range is similar to a disk already, we can find the smallest disk containing the numerical range, which is $\mathcal{D}(14.9321, 13.8522)$ (cyan, dashed), and use Faber polynomials to approximate the best polynomial which are equivalent to the Taylor series for a disk. The lower right plot shows the 2-norm of the errors of the 2-norm optimal Krylov approximation (green, dotted) and the error from using the Faber polynomial approximation of $R(z)$ on the disk containing the numerical range (blue, solid). To bound the error according to (3.12), we need to also multiply the error of the Faber polynomial approximation by $\kappa(\mathbf{S})^{1/2}(1 + \sqrt{2}) \approx 1.053e + 3$. While the error bound is a large over-approximation, it does indicate convergence.

In Figure 3.3, the randomly generated values for $\mathbf{R}$, $\mathbf{b}$, $D(z)$, and $N(z)$ are all equivalent to those used to generate Figure 3.1 where now $\mathbf{A} = \mathbf{R} + 25\mathbf{I}$. The condition number of the eigenvector matrix is the same, and $\kappa(D(\mathbf{A})^* D(\mathbf{A}))^{1/2}$ is approximately 24.6. The top plots show the 2-norm of the residuals and errors at each iteration. Arnoldi-OR, Arnoldi-FA, GMRES applied to a partial fraction decomposition, and the 2-norm optimal Krylov approximation converge in about 36 iterations. GMRES applied to $D(\mathbf{A})$ converges in about 95 iterations.

The lower left plot shows that there is a sizable distance between the roots of $D(\mathbf{A})$ (red, 'o's) and both the boundary of the numerical range of $\mathbf{A}$ (black, solid) and the boundary of the smallest disk containing $W(\mathbf{A})$, $\mathcal{D}(24.9321, 13.8522)$ (cyan, dashed). The roots of $R(z)$ (red, 'o's) are even more distant from the eigenvalues (blue, 'x's). Just like in Figure 3.2, we can use (3.12) to bound the error. The lower right plot shows the error of the 2-norm optimal Krylov approximation (green, dotted) and the truncated Faber series approximation for $R(z)$

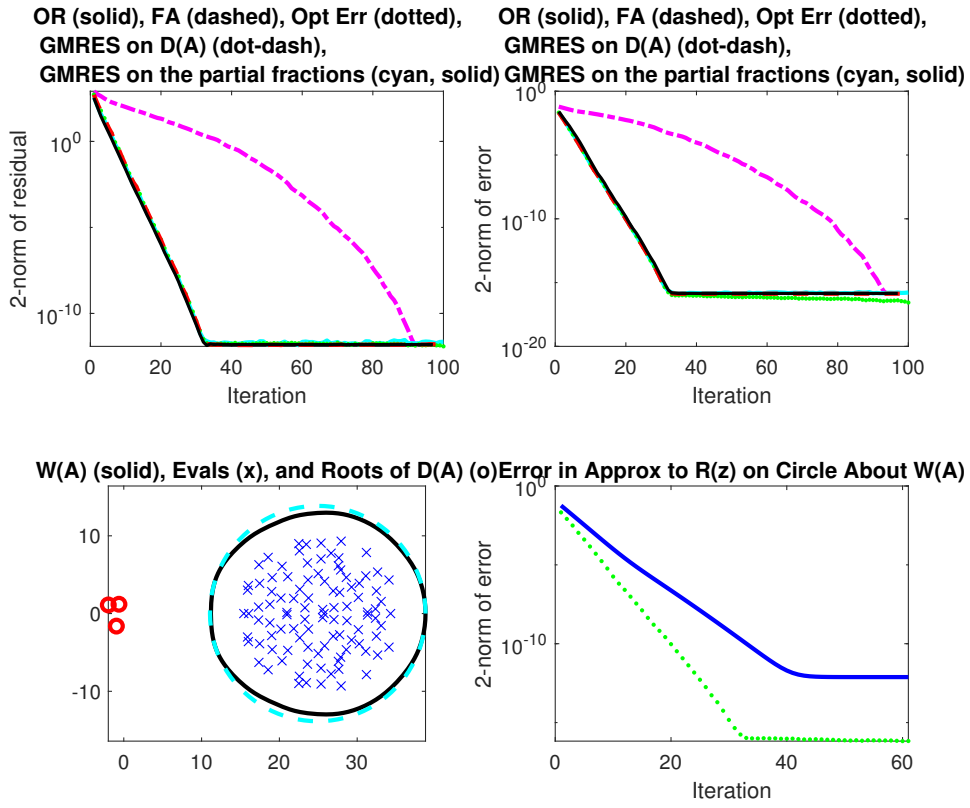


Figure 3.3: $\mathbf{A} = \text{randn}(100) + 25\mathbf{I}$, $\mathbf{b} = \text{randn}(100, 1)$, $N(z)$ random quadratic, $D(z)$ random cubic, generated in MATLAB with the seed rng("default") in the order presented, and are equivalent to the randomly generated values used in Figure 3.1. Top plots show the 2-norms of the residuals and 2-norms of the errors for Arnoldi-OR (black, solid), Arnoldi-FA (red, dashed), GMRES applied to $D(\mathbf{A})$ (magenta, dot-dash), GMRES applied to a partial fraction decomposition (cyan, solid), and the 2-norm optimal Krylov approximation from $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ (green, dotted). Lower left plot shows eigenvalues of $\mathbf{A}$ (blue, 'x's), $\partial W(\mathbf{A})$ (black, solid), and poles of $R(z)$ (red, 'o's). Lower right plot compares the errors of the 2-norm optimal Krylov approximation from $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ (green, dotted) and a numerical approximation of $\min_{p_{k-1} \in \mathcal{P}_{k-1}} \sup_{z \in \mathcal{D}(24.9, 13.9)} |R(z) - p_{k-1}(z)|$ (blue, solid) which can be used in the error bound (3.12).

on the smallest disk containing the numerical range (blue, solid). To use the bound (3.12), we must multiply the truncated Faber series approximation by $\kappa(\mathbf{S})^{1/2}(1+\sqrt{2}) \approx 7.15e{+}01$. This time, the error bound is closer to the optimal error reduction.

Figure 3.4 shows comparisons of 2-norms of the residuals and 2-norms of the errors

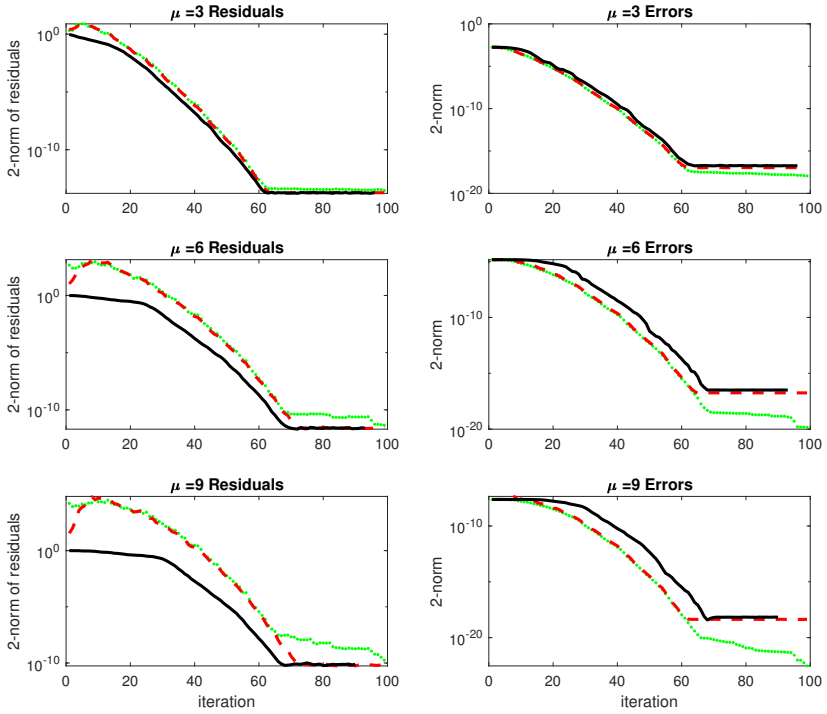


Figure 3.4: $\mathbf{A} = \text{randn}(100) + 15\mathbf{I}$, $\mathbf{b} = \text{randn}(100, 1)$, $N(z)$ is the identity, and each row in the figure tests a different $D(z)$ with μ random roots. The random values are generated in MATLAB by setting the random number generator seed to zero so that $\mathbf{A}$ and $\mathbf{b}$ are the same as in Figure 3.2, and then generating the denominator polynomial polynomials with $\mu = 3$, $\mu = 6$, and then $\mu = 9$. Each row shows the 2-norms of the residuals on the left and 2-norms of the errors on the right for Arnoldi-OR (black, solid), Arnoldi-FA (red, dashed), and the 2-norm optimal approximation from $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ (green, dotted).

for Arnoldi-OR (black, solid), Arnoldi-FA (red, dashed), and the 2-norm optimal Krylov approximation (green, dotted) for $R(\mathbf{A})$ with denominators with differing maximum degree polynomials. The matrix $\mathbf{R}$ and vector $\mathbf{b}$ are equivalent to those used in the rest of this section, and $\mathbf{A} = \mathbf{R} + 15\mathbf{I}$ like in Figure 3.2. Unlike in the previous examples from this section, the numerator, $N(\mathbf{A})$, is equal to the identity. The plots in each row of the figure contain denominators with a randomly generated coefficient and μ randomly generated roots. The plots in the left column show the 2-norms of the residuals, and the plots in

the right column show the 2-norms of the errors. We see that as the maximum degree of the denominator increases, there is a widening gap between the residuals and errors of Arnoldi-OR (black, solid) compared to the residuals and errors of Arnoldi-FA (red, dashed) and 2-norm optimal Krylov approximation (green, dotted). Interestingly, the residuals of the Arnoldi-FA and the 2-norm optimal Krylov approximation have a maximum value on the order of $e+00$ when $\mu = 3$, $e+03$ when $\mu = 6$, and $e+05$ when $\mu = 9$, whereas on the same iteration that the maximum 2-norm of the residual is reached, the 2-norm of the error is on the order of $e-03$ when $\mu = 3$, $e-05$ when $\mu = 6$, and $e-08$ when $\mu = 9$. As the degree of the denominator increases, $\kappa(\mathbf{S}) = \kappa(D(\mathbf{A})^* D(\mathbf{A}))$ increases. At $\mu = 3$ $\kappa(\mathbf{S}) \approx 356.6$, at $\mu = 6$ $\kappa(\mathbf{S}) \approx 5.4\text{e+}4$, and at $\mu = 9$ $\kappa(\mathbf{S}) \approx 2.333\text{e+}6$. However, the errors of all three methods converge to machine precision in a similar amount of time regardless of the maximum degree of the denominator polynomial, about 60 iterations.

Other parameters tested for an effect on the convergence of the three methods were varying the maximum degree of the numerator polynomial and varying the dimension of the randomly generated $\mathbf{A}$. Increasing the maximum degree of the numerator affects residuals and errors of the first couple of iterations, but not the overall convergence. Increasing the dimension of the randomly generated matrix has no clear effect on the convergence of Arnoldi-OR, Arnoldi-FA, and the 2-norm optimal Krylov approximation as long as the roots of the polynomial $D(z)$ lie outside of the numerical range of $\mathbf{A}$.

3.3.2 Grcar Matrix

The Grcar matrix is a Toeplitz matrix, and a Toeplitz matrix is a diagonally constant matrix. Non-Hermitian Toeplitz matrices are one of the best understood families of highly non-normal matrices [108, Ch. 7]. The Grcar matrix is a non-Hermitian Toeplitz matrix with -1's on the first sub-diagonal, 1's on the main diagonal, and 1's on the next three super-diagonals. In MATLAB, to define an n dimensional square Grcar matrix we use the command `gallery('grcar', n)`. The eigenvalues of a Grcar matrix are all highly ill-

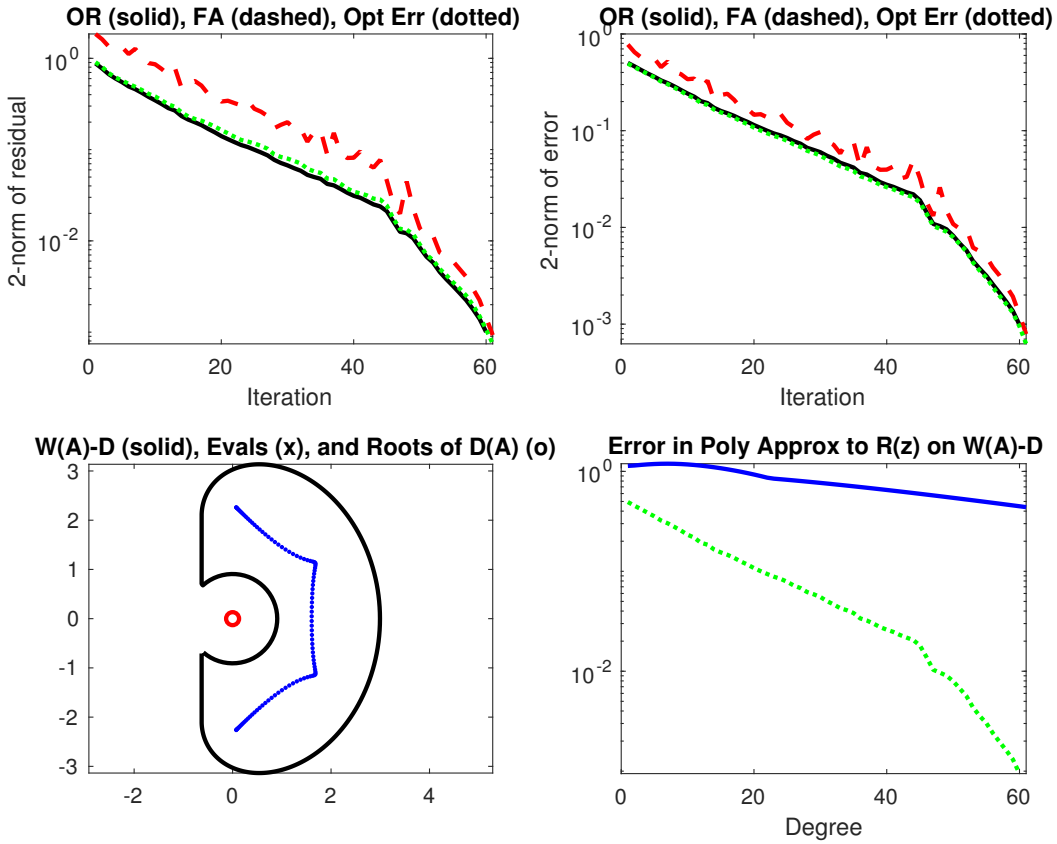


Figure 3.5: $\mathbf{A}$ = `gallery('grcar',100)`, $\mathbf{b} = randn(100)$, $N(z) = 1$, and $D(z) = z$. The vector $\mathbf{b}$ is generated in MATLAB after those used in Figures 3.1, 3.2, and 3.3. Top plots show the 2-norms of the residuals and the 2-norms of the errors for Arnoldi-OR (solid, black), Arnoldi-FA (dashed, red), and the 2-norm optimal Krylov approximation from $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ (dotted, green). Lower left plot shows eigenvalues of $\mathbf{A}$ (blue, 'x's), the boundary of $\Omega = W(\mathbf{A}) \backslash \mathbb{D}(0, 1/w(\mathbf{A}^{-1}))$ (black, solid), and the pole of $R(z)$ (red, 'o'). Lower right plot compares the errors of the 2-norm optimal Krylov approximation (green, dotted) and a numerical approximation of $\min_{p_{k-1} \in \mathcal{P}_{k-1}} \sup_{z \in \Omega} |R(z) - p_{k-1}(z)|$ (blue, solid) which can be used in the error bound (3.15).

conditioned, so bound (3.11), which uses $\kappa(\mathbf{V})$, is not useful. For example, when $n = 100$, the condition number of the eigenvector matrix of the Grcar matrix is about $5.1e{+}17$.

In Figure 3.5, $\mathbf{A}$ is the 100 by 100 Grcar matrix, $\mathbf{b}$ is randomly generated using the command `randn(100,1)` in MATLAB and is generated after the coefficient and roots of the denominator polynomial in Figures 3.1, 3.2, and 3.3. For this example $N(\mathbf{A}) = \mathbf{I}$

and $D(\mathbf{A}) = \mathbf{A}$. For this rational function, $R(\mathbf{A}) = \mathbf{A}^{-1}$, Arnoldi-OR is equivalent to GMRES. The top plots show the 2-norms of the residuals, $\|N(\mathbf{A})\mathbf{b} - D(\mathbf{A})\mathbf{x}_k\|_2$, (left) and the 2-norms of the errors, $\|D(\mathbf{A})^{-1}N(\mathbf{A})\mathbf{b} - \mathbf{x}_k\|_2$, (right) of Arnoldi-OR (black, solid), Arnoldi-FA (red, dashed), and the 2-norm optimal Krylov approximation (green, dotted). Arnoldi-OR, Arnoldi-FA, and the projected solution all converge in 94 iterations, and after iteration 60, the residuals and errors of all three methods are similar. The lower left plot shows the eigenvalues of $\mathbf{A}$ (blue, 'x's), the boundary of the set Ω equal to the numerical range of $\mathbf{A}$ without a disk centered at zero with a radius equal to the inverse of the numerical radius of $\mathbf{A}^{-1}$ (solid, black), and the pole of $R(z)$ at the origin (red, 'o'). Before removing the disk $\mathcal{D}(0, 1/w(\mathbf{A}^{-1}))$, the numerical range of $\mathbf{A}$ contained the origin, so the bound (3.12) would be infinite and not useful. Instead, we use bound (3.15) with $m = 1$, which bounds K by $3+2\sqrt{3}$. For the Grcar matrix with $n = 100$ and $D(\mathbf{A}) = \mathbf{A}$, $\kappa(\mathbf{A}^*\mathbf{A})^{1/2}$ is approximately 3.6. To approximate the best polynomial approximation on the set $W(\mathbf{A})/\mathcal{D}(0, 1/w(\mathbf{A}^{-1}))$, we use the Schwarz-Christoffel package to compute the conformal map from the exterior of the unit disk to the exterior of this set and derive the Faber series [38]. In the lower right plot, we compare the errors of the truncated Faber series (blue, solid) and the 2-norm optimal Krylov approximation (green, dotted). To calculate the error bound (3.15), we must multiply the errors of the truncated Faber series by $\kappa(\mathbf{S})^{1/2}(3 + 2\sqrt{3}) \approx 2.3e + 01$. While the error bound (3.15) is an overestimate, the negative slope does predict eventual convergence, which no other error bounds in Section 3.2 would do for the 100 dimensional Grcar matrix.

3.3.3 Matrix Exponential

In Section 3.3.1, we noticed that when the poles of $R(z)$ were farther away from both the eigenvalues and numerical range of $\mathbf{A}$, then the residuals and errors of Arnoldi-OR, Arnoldi-FA, and the 2-norm optimal Krylov approximation all behaved similarly. Many functions, including $\exp(\mathbf{A})$, can be approximated by a rational function with poles far

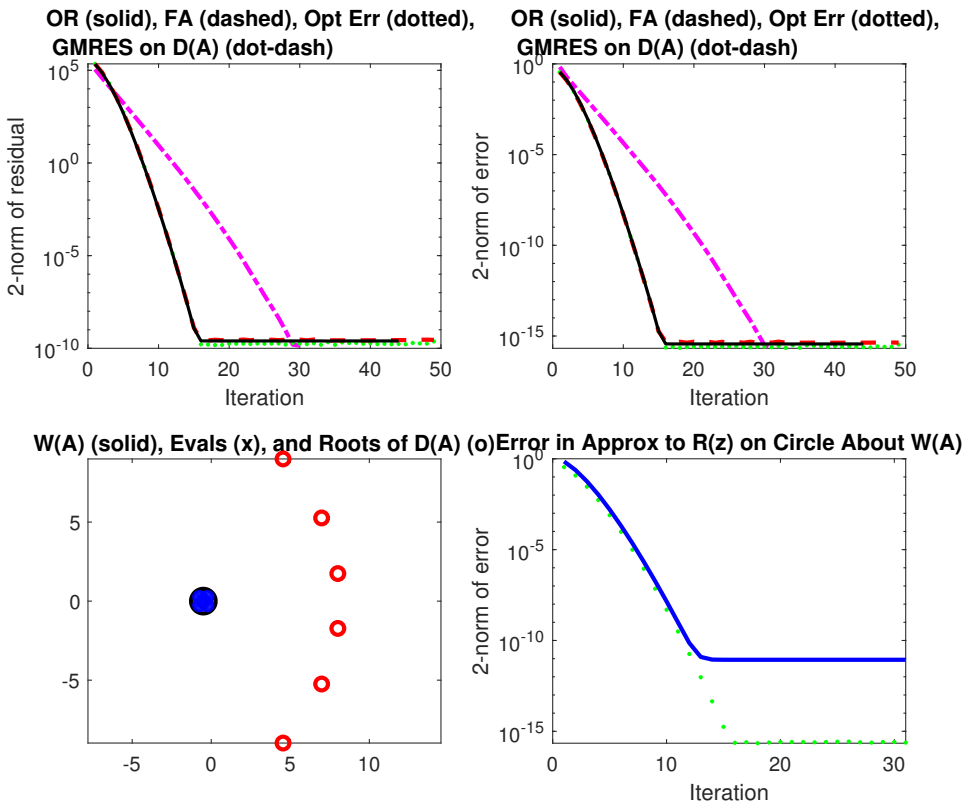


Figure 3.6: $\mathbf{A}$ = transient_demo(50), $\mathbf{b}$ = randn(50,1), and we solve for $\mathbf{x} = \exp(\mathbf{A})\mathbf{b}$ where D and N have degree 6 and are calculated using RKToolkit [8]. Top plots show the 2-norms of the residuals and the 2-norms of the errors of a rational function approximation of $\exp(\mathbf{A})\mathbf{b}$ for Arnoldi-OR (black, solid), Arnoldi-FA (red, dashed), GMRES applied to $D(\mathbf{A})$ (magenta, dot-dash), and the 2-norm optimal Krylov approximation from $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ (green, dotted). Lower left plots show eigenvalues of $\mathbf{A}$ (blue, 'x's), $\partial W(\mathbf{A})$ (black, solid), and the poles of $R(z)$ (red, 'o's). Lower right plots compares the 2-norm of the errors of the 2-norm optimal Krylov approximation (green, dotted) and a numerical approximation of $\min_{p_{k-1} \in \mathcal{P}_{k-1}} \max_{z \in \mathcal{W}(\mathbf{A})} |R(z) - p_{k-1}(z)|$ (blue, solid) which can be used in the error bound (3.12).

from the eigenvalues of $\mathbf{A}$.

Let $\mathbf{A}$ be the `transient_demo(50)` matrix from the eigtool package [120]. The condition number of the eigenvector matrix is about $6.4e+04$, indicating that `transient_demo(50)` is a highly non-normal matrix and that bound (3.11) is less likely to be useful. Let $\mathbf{b}$ be a random vector of length 50 sampled from the standard normal distribution and normalized

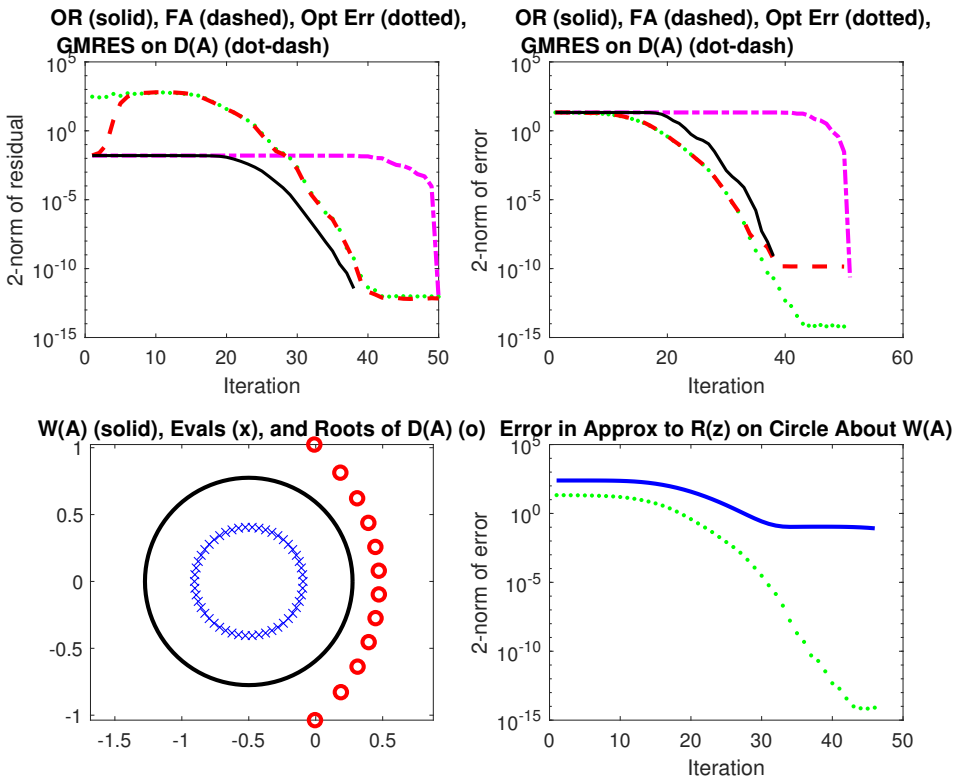


Figure 3.7: $\mathbf{A}$ = transient_demo(50) and $\mathbf{b}$ = randn(50,1) are equivalent to those used in Figure 3.6. We solve for $\mathbf{x} = \exp(20\mathbf{A})\mathbf{b}$ where D and N have degree 12 and are calculated using RKToolkit [8]. Top plots show the 2-norms of the residuals and the 2-norms of the errors of a rational function approximation of $\exp(\mathbf{A})\mathbf{b}$ for Arnoldi-OR (black, solid), Arnoldi-FA (red, dashed), GMRES applied to $D(\mathbf{A})$ (magenta, dot-dash), and the 2-norm optimal Krylov approximation from $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ (green, dotted). Lower left plots show eigenvalues of $\mathbf{A}$ (blue, 'x's), $\partial W(\mathbf{A})$ (black, solid), and the poles of $R(z)$ (red, 'o's). Lower right plots compares the 2-norm of the errors of the 2-norm optimal Krylov approximation (green, dotted) and a numerical approximation of $\min_{p_{k-1}\in\mathcal{P}_{k-1}} \max_{z\in\mathcal{W}(\mathbf{A})} |R(z) - p_{k-1}(z)|$ (blue, solid) which can be used in the error bound (3.12).

for convenience. This vector was randomly generated after the right-hand side vector in Section 3.3.2. We calculate a rational function approximation to $\exp(t\mathbf{A})$ by using the `rkfit` function from the Rational Krylov Toolkit (RKToolkit) [8], where $t \geq 1$ is a time parameter. When $t = 1$, the approximation $R(z)$ has a maximum degree of six for both the numerator and denominator polynomials. We compared MATLAB's built in function

`expm(A)*b` and the rational approximation output by `rkfit` with multiprecision arithmetic to find that the approximation matched $\exp(\mathbf{A})\mathbf{b}$ to near machine precision. Further, we find the denominator polynomial is well-conditioned since $\kappa(D(\mathbf{A})^* D(\mathbf{A}))^{1/2}$ is approximately 2.17.

In Figure 3.6, the top plots show the 2-norms of the residuals at each iteration, $\|N(\mathbf{A})\mathbf{b} - D(\mathbf{A})\mathbf{x}_k\|_2$, and the 2-norms of the errors at each iteration, $\|R(\mathbf{A})\mathbf{b} - \mathbf{x}_k\|_2$, for Arnoldi-OR (black, solid), Arnoldi-FA (red, dashed), GMRES applied to $D(\mathbf{A})$ (magenta, dot-dash), and the 2-norm optimal Krylov approximation (green, dotted). Arnoldi-OR, Arnoldi-FA, and the 2-norm optimal Krylov approximation all converge in fifteen iterations, whereas GMRES applied to $D(\mathbf{A})$ converges in twenty-nine iterations.

The bottom left plot shows the eigenvalues of $\mathbf{A}$ (blue, 'x's), the boundary of the numerical range of $\mathbf{A}$ (black, solid), and poles of $R(z)$ (red, 'o's). Since the numerical range of $\mathbf{A}$ is a disk, the conformal map from the exterior of the unit disk to the exterior of the numerical range is simple to calculate, and the Faber polynomials to approximate the best polynomial approximation to $R(z)$ reduce to a Taylor series. The bottom right plot compares the errors of the truncated Faber series approximation of $R(z)$ (blue, solid) to the errors of the 2-norm optimal Krylov approximation (green, dotted), and shows that they are very similar in this example until the error of the truncated Faber series levels off. While the error of the truncated Faber series levels off above machine precision, it does so only four iterations before all three methods converge.

It is harder to get a good approximation with a rational function when trying to approximate $\exp(t\mathbf{A})\mathbf{b}$ for $t \gg 1$. The rational function approximation requires higher degree polynomials in the numerator and denominator, which increases the likelihood that we will be working with a more ill-conditioned problem. Approximating these higher degree rational functions with polynomials of degree $k-1$ also gets more difficult. For more ways to approximate $\exp(\mathbf{A})\mathbf{b}$, see Moler and Van Loan (2003) [82].

In Figures 3.7 and 3.8 we show results for approximating $R(t\mathbf{A})\mathbf{b}$, where $\mathbf{A}$ and $\mathbf{b}$ are

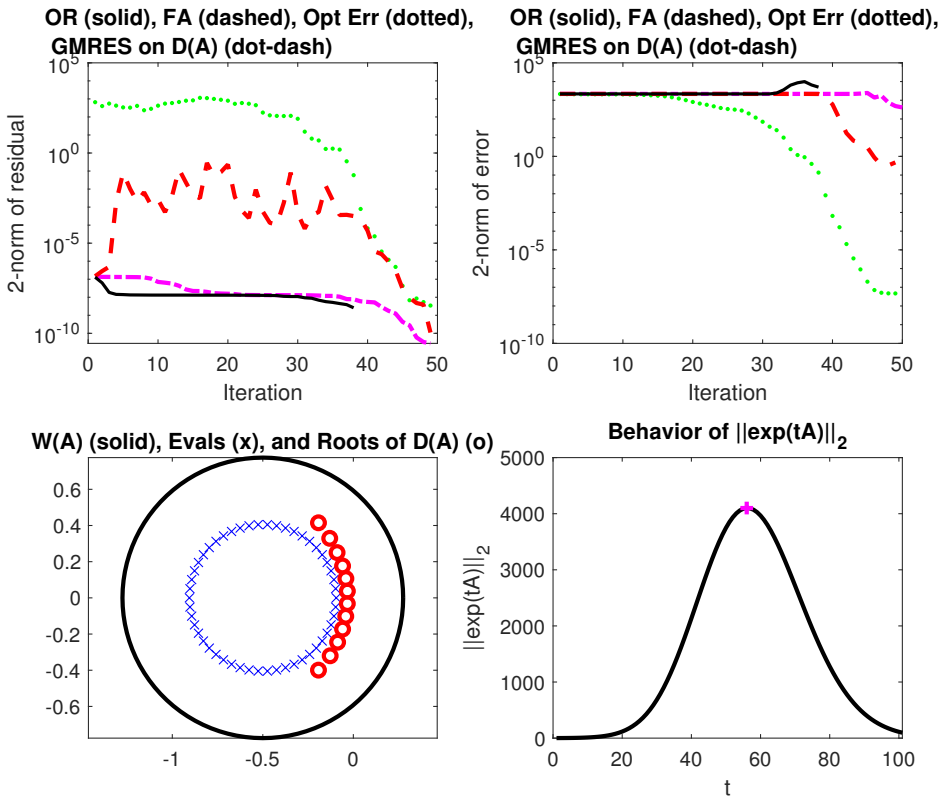


Figure 3.8: $\mathbf{A} =$ transient_demo(50) and $\mathbf{b} =$ randn(50,1) are equivalent to those used in Figure 3.6. We solve for $\mathbf{x} = \exp(56\mathbf{A})\mathbf{b}$ where D and N have degree 12 and are calculated using RKToolkit [8]. Top plots show the 2-norms of the residuals and the 2-norms of the errors Arnoldi-OR (black, solid), Arnoldi-FA (red, dashed), GMRES applied to $D(\mathbf{A})$(magenta, dot-dash), and the 2-norm optimal Krylov approximation from $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ (green, dotted). Lower left plot shows eigenvalues of $\mathbf{A}$ (blue, 'x's), $\partial W(\mathbf{A})$ (black, solid), and poles of $R(z)$ (red, 'o's). Lower right plot shows $\| \exp(t\mathbf{A})\|_2$ (black, solid), and $\| \exp(56\mathbf{A})\|_2$ (magenta, '+').

the same as in Figure 3.6, and $R(t\mathbf{A})\mathbf{b}$ is a rational approximation to $\exp(t\mathbf{A})\mathbf{b}$ with twelve degree numerator and denominator polynomials for $t = 20$ in Figure 3.7 and $t = 56$ in Figure 3.8. When $t = 20$, the rational approximation computed with the `rkfit` function from the RKToolkit has an error of $3.2e$-10. When $t = 56$, the rational approximation computed with the `rkfit` function has an error of $3.3e$-8. Both of these errors were computed by calculating the 2-norm of the difference between `expm(t*A)*b` (using MATLAB's built-in

function) and the rational function output from `rkfit`.

The top plots show the 2-norms of the residuals and the 2-norms of the errors for Arnoldi-OR (black, solid), Arnoldi-FA (red, dashed), GMRES applied to $D(\mathbf{A})$ (magenta, dot-dash), and the 2-norm optimal Krylov approximation (green, dotted). Since the dimension of $\mathbf{A}$ is 50, and the maximum degrees of $D(z)$ and $N(z)$ are 12, Arnoldi-OR can only run for 39 iterations. During those first 39 iterations, Arnoldi-OR necessarily has the smallest residuals at each iteration. When $t = 20$, at iteration 38 Arnoldi-OR reached a residual value on the order of $1e$-12, and for $t = 56$ Arnoldi-OR reached a minimum residual value on the order of $1e$-10. The 2-norm of the error is necessarily the smallest for the optimal Krylov approximation in the 2-norm (green, dotted). When $t = 20$ the 2-norm optimal Krylov approximation converges around iteration 45, the remaining methods require running 50 iterations before converging. When $t = 56$, Arnoldi-FA, GMRES applied to $D(\mathbf{A})$, and the 2-norm optimal Krylov approximation converge at iteration 50. From the bottom left plots of both Figures 3.7 and 3.8, we see that when $t = 20$ the poles of $D(z)$ (red, 'o's) are much closer to the boundary of the numerical range (black, solid) than when $t = 1$ in Figure 3.6. However, when $t = 56$ the poles are located entirely within the numerical range and all 12 of the poles (red, 'o's) are close to ill-conditioned eigenvalues (blue, 'x's).

The bottom right plot of Figure 3.7 compares the maximum error of the truncated Faber series approximation of $R(z)$ (blue, solid) to the 2-norm optimal Krylov approximation (green, dotted). Since the numerical range of $\mathbf{A}$ is a disk, the Faber series still reduces to the Taylor series of $R(z)$, but since the poles of $R(z)$ are much closer to the boundary of the numerical range the error of the truncated Faber series has a wider gap from the 2-norm optimal Krylov approximation than in Figure 3.6.

The bottom right plot of Figure 3.8 shows $\|\exp(t\mathbf{A})\|_2$ (black, solid) for $t \in [1, 100]$, and $\|\exp(56\mathbf{A})\|_2 \approx 4.1e+03$ (magenta, '+'). We see in this plot that despite having only eigenvalues with real part less than zero, $\|\exp(t\mathbf{A})\|_2$ grows significantly before converging towards zero as the eigenvalues indicate. This transient growth is one of the properties that

makes highly non-normal matrices so interesting. However, despite the norm of $\|\exp(t\mathbf{A})\|_2$ converging back to small values, as t increases the rational approximation of $\exp(t\mathbf{A})$ continues to grow more ill-conditioned.

3.4 Remarks

Arnoldi-OR uses $\max\{\deg(D(\mathbf{A})), \deg(N(\mathbf{A}))\} - 1$ extra steps of the Arnoldi algorithm in order to find the optimal (in $D(\mathbf{A})^* D(\mathbf{A})$-norm) approximation to $D(\mathbf{A})^{-1} N(\mathbf{A})\mathbf{b}$ from the Krylov space $\text{span}(\mathbf{b}, \mathbf{Ab}, \ldots, \mathbf{A}^{k-1}\mathbf{b})$. To do this requires solving a $k+\max\{\deg(D(\mathbf{A})), \deg(N(\mathbf{A}))\}$ by k least squares problem. The least squares problem can be solved via QR factorization with Givens rotations—applying previous rotations to just the last column of the newly formed matrix and choosing new rotations to eliminate entries below the diagonal in column k.

While Arnoldi-OR generates approximations that are provably optimal in the 2-norm of the residual, Arnoldi-FA appears to perform similarly. Further, after some lead-in time, the 2-norm of the error for Arnoldi-FA appears to be at least as good as the 2-norm of the error of Arnoldi-OR. However, there is not currently an explanation of why this phenomenon occurs. In general, our numerical experiments indicate that the most important factors for convergence are the proximity of the poles of $R(z)$ to the eigenvalues of $\mathbf{A}$ and having a well conditioned denominator polynomial.

Further, the *a priori* error bounds for Arnoldi-OR are often large overestimates, but they do indicate convergence in cases where previous bounds could not. For example, the error bound in (3.15) indicates convergance for $R(\mathbf{A})\mathbf{b}$ even when the poles of the rational function are located within the numerical range of $\mathbf{A}$. While Arnoldi-OR may not be a first choice algorithm for computing the action of a rational function of a matrix on a vector, it has the potential to improve our theoretical understanding of Arnoldi-FA.

Overall Arnoldi-OR shows theoretical promise as an algorithm, and may be able to provide convergence insight for other algorithms. However, practitioners are not likely to replace Arnoldi-FA with Arnoldi-OR.

Chapter 4

DEPARTURES FROM NORMALITY

There are many equivalent definitions of a *normal matrix*, including a matrix satisfying $\mathbf{A}\mathbf{A}^* = \mathbf{A}^*\mathbf{A}$ and a square matrix with a complete set of orthonormal eigenvectors. A *non-normal matrix* is any matrix not satisfying these definitions. Some research involving non-normal matrices focused on clarifying how algorithms behave differently for normal and non-normal matrices, especially for the connection between (non-)normality and eigenvalue problems [41].

Let $\mathscr{N}$ be the set of all normal matrices. One might measure how non-normal the matrix $\mathbf{A}$ is using

$$\|\mathbf{A}\mathbf{A}^* - \mathbf{A}^*\mathbf{A}\|_2, \text{ or}$$

$$\min_{\mathbf{N}\in\mathscr{N}} \|\mathbf{A} - \mathbf{N}\|_2.$$

The second measure of non-normality can be reliably calculated using work by Gabriel (1979) [49] and Ruhe (1987) [94]. A third way to measure the non-normality of a diagonalizable matrix is to use the condition number of an eigenvector matrix, $\kappa(\mathbf{V})$, and preferably the best-conditioned eigenvector matrix. In 1960, Bauer and Fike began using $\kappa(\mathbf{V})$ to measure the sensitivity of the eigenvalues of a matrix to perturbations [4].

While it is convenient to use a single number to describe how non-normal a matrix is, other ways to characterize it may prove useful in applications. For example, the *numerical range* of a normal matrix is the convex hull of the eigenvalues. However, the *numerical range* of a non-normal matrix is a convex set containing the convex hull of the eigenvalues and possibly much more. The *ϵ-pseudospectrum* of a normal matrix is the union of disks of radius ϵ about each eigenvalue, but for a non-normal matrix the *ϵ-pseudospectrum* is a subset of

the union of disks of radius $\epsilon\kappa(\mathbf{V})$ about the eigenvalues that contains the union of disks of radius ϵ about each eigenvalue. The *K-spectral set* of a matrix $\mathbf{A}$ is any set Ω in the complex plane containing the eigenvalues of $\mathbf{A}$ with the property that $\|f(\mathbf{A})\|_2 \leq K \sup_{z\in\Omega} |f(z)|$ for all analytic functions f in Ω. The numerical range and the ϵ-pseudospectrum of a matrix are both K-spectral sets but with different values of K.

Section 4.1 discusses how a diagonalizable matrix's non-normality is measured using an eigenvector matrix's condition number. Section 4.2 discusses the numerical range of a matrix and the bounds on $\|\mathbf{A}^k\|_2$ and $\|\exp(\mathbf{A})\|_2$ that can be derived from the numerical range. Section 4.3 discusses pseudospectral sets, their properties, and their usages when calculating bounds on $\|f(\mathbf{A})\|_2$. Finally, Section 4.4 introduces K-spectral sets and two ways to calculate an upper bound on K. In Chapter 5, we will build on the K-spectral set theory presented and explain our contributions to the field.

4.1 Bauer-Fike Theorem and $\kappa(\mathbf{V})$

In 1960, Bauer and Fike introduced the condition number of an eigenvector matrix, $\kappa(\mathbf{V}) = \|\mathbf{V}\|_2\|\mathbf{V}^{-1}\|_2$, as a way of summarizing how sensitive to perturbations the eigenvalues of a diagonalizable matrix will be. For normal matrices, the condition number of the best-conditioned eigenvector matrix is equal to one. For diagonalizable non-normal matrices, $\kappa(\mathbf{V})$ of the best conditioned eigenvector matrix may be arbitrarily large. For non-diagonalizable, non-normal matrices, $\kappa(\mathbf{V})$ is considered to be infinite.

The Bauer-Fike Theorem, first published in 1960, uses the condition number of an eigenvector matrix to bound the maximum perturbation of eigenvalues of a matrix $\mathbf{A}$ using any perturbation matrix $\mathbf{E}$ [4].

Theorem 4.1:
Suppose $\mathbf{A} \in \mathbb{C}^{n\times n}$ is diagonalizable so that $\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1}$. Let μ be an eigenvalue of $\mathbf{A}+\mathbf{E}$. Then there exists λ, an eigenvalue of $\mathbf{A}$, such that

$$|\lambda - \mu| \leq \kappa(\mathbf{V})\|\mathbf{E}\|_2.$$

Since $\kappa(\mathbf{V})$ quantifies how far the eigenvalues of a diagonalizable matrix can be perturbed, how non-normal a matrix is may be measured by $\kappa(\mathbf{V})$.

4.2 The Numerical Range

The numerical range of a matrix, $W(\mathbf{A})$ has long been used to yield more information about a matrix than the eigenvalues alone can say. Define the 2-norm inner product as $\langle \mathbf{y}, \mathbf{Ax} \rangle := \mathbf{y}^* \mathbf{Ax}$.

Definition 4.1. *The numerical range, $W(\mathbf{A})$ of $\mathbf{A} \in \mathbb{C}^{n \times n}$ is the set*

$$W(\mathbf{A}) = \{ \langle \mathbf{q}, \mathbf{Aq} \rangle : \mathbf{q} \in \mathbb{C}^n, \langle \mathbf{q}, \mathbf{q} \rangle = 1 \}.$$

This set is always convex [105, 68], contains the eigenvalues of the matrix, and can be used to understand how algorithms will run differently on non-normal matrices compared to normal matrices [74].

For a normal matrix, the numerical range is equivalent to the convex hull of the eigenvalues. For a non-normal matrix, the numerical range is a convex set containing the convex hull of eigenvalues. Figure 4.1 contains an example comparing the numerical ranges of a normal matrix and a non-normal matrix with the same eigenvalues. How much the numerical range extends beyond the convex hull of the eigenvalues can be a heuristic for estimating how non-normal a matrix is.

Bounding $\|f(\mathbf{A})\|_2$ when $\mathbf{A}$ is non-normal is interesting because of a phenomenon known as transient growth, which is not explained by the eigenvalues. Two constant values that are derived from the numerical range and can be used to bound $\|f(\mathbf{A})\|_2$ are the numerical abscissa and numerical radius. The numerical abscissa of a matrix is $\rho(\mathbf{A}) := \max(\mathrm{Re}(W(\mathbf{A})))$, and the numerical radius is $w(\mathbf{A}) := \max |W(\mathbf{A})|$. For example, the instantaneous slope of the solution to the continuous time dynamical system $\mathrm{d}\mathbf{x}/\mathrm{d}t = \mathbf{Ax}(t)$ is the numerical

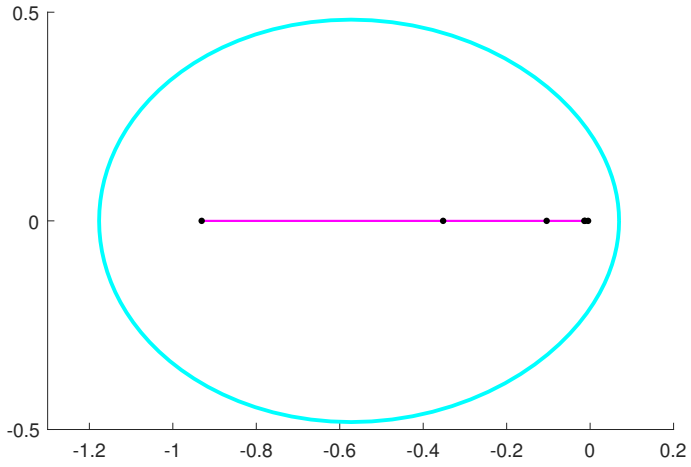


Figure 4.1: The numerical range of a normal and a non-normal matrix with the same eigenvalues. The black dots are the eigenvalues of both matrices, the fuschia line along the real axis indicates the numerical range of the normal matrix, and the cyan curve is the numerical range of the non-normal matrix.

abscissa $\rho(\mathbf{A})$ where

$$\frac{\mathrm{d}}{\mathrm{d}t}\|\exp(t\mathbf{A})\|_2\Big|_{t=0} = \lim_{t\downarrow 0} t^{-1}\log\|\exp(t\mathbf{A})\|_2 = \rho(\mathbf{A}), \tag{4.1}$$

and the numerical abscissa is also in the bound

$$\|\exp(t\mathbf{A})\|_2 \leq \exp(t\rho(\mathbf{A})) \qquad \forall t \geq 0. \tag{4.2}$$

The powers of $\mathbf{A}$ are bounded using the numerical radius $w(\mathbf{A})$ by

$$\|\mathbf{A}^k\|_2 \leq 2w(\mathbf{A}^k) \leq 2w(\mathbf{A})^k \tag{4.3}$$

[74, §1.5]. The numerical range can tell us much about a matrix, and it can be used to

bound norms of functions of matrices.

4.3 Pseudospectra

Pseudospectra are an in-depth way to characterize the non-normality of a matrix. Pseudospectral sets, $\Lambda_\epsilon(\mathbf{A})$, are regions inside contours of the resolvent norm of $\mathbf{A}$ in the complex plane. In Figure 4.2 is an example of pseudospectral sets for the Tuesday Lake matrix, which contains the constant linear coefficients of a dynamical system modeling the effects of introducing a new species of piscivorous fish into Tuesday Lake, Michigan in 1984 [84].

Definition 4.2. *Let* $\mathbf{A} \in \mathbb{C}^{N\times N}$ *and* $\epsilon > 0$ *be arbitrary. The* ϵ*-pseudospectrum,* $\Lambda_\epsilon(\mathbf{A})$*, is the set of* $z \in \mathbb{C}$ *such that*

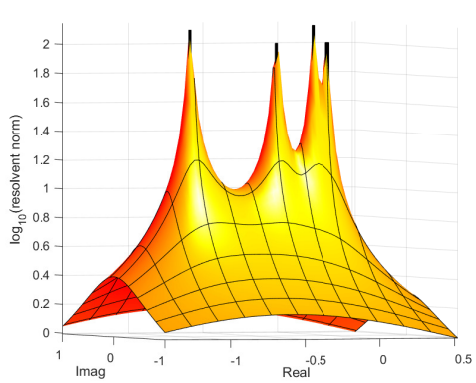
$$\|(z\mathbf{I} - \mathbf{A})^{-1}\|_2 > \epsilon^{-1}.$$

One can equivalently define the ϵ-pseudospectrum as follows.

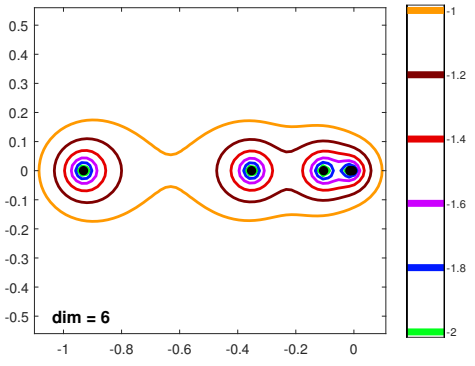
Definition 4.3. *The* ϵ*-pseudospectrum is the set of numbers that are eigenvalues of some perturbed matrix* $\mathbf{A} + \mathbf{E}$ *with* $\|\mathbf{E}\|_2 < \epsilon$*.*

This definition quantifies how much the eigenvalues of a matrix change when the matrix is perturbed by a matrix of norm ϵ. Usually pseudospectra are calculated using the 2-norm, however both definitions of these sets may be generalized for other matrix norms. In practice, definition 4.2 is used to compute pseudospectral sets of a matrix, for instance with the Eigtool package [120], which was used to generate the plots in Figure 4.2.

Pseudospectral sets are well known in part due to Trefethen and Embree's 2005 book *Spectra and Pseudospectra* [108]. In the introduction of a 1991 paper, Trefethen describes the research of the preceeding 40 years as focusing on "how eigenvalues change under perturbations rather than on the exploitation of information that goes beyond eigenvalues" [107]. In the 1980's eigenvalues were trusted for all sorts of analyses, including for analysis of highly non-normal matrices. This contradiction between how analysis was carried out in practice, and the well-known theory that non-normal matrix behavior is not characterized



(a) A 3D plot of the inverse of the resolvent norm, $\|(zI - \mathbf{T})^{-1}\|_2$, of the Tuesday Lake matrix $\mathbf{T}$ [84]. The singularities of the plot are at the eigenvalues of $\mathbf{T}$.



(b) A plot of pseudospectral sets, $\Lambda_\epsilon(\mathbf{T})$, for different values of ϵ. Inside each simple closed curve of $\Lambda_\epsilon(\mathbf{T})$ are one or more eigenvalues of $\mathbf{T}$. The different values of ϵ are on a log_{10} scale visible to the right. Each color represents the boundary of $\Lambda_\epsilon(\mathbf{A})$ for a different value of ϵ.

Figure 4.2: An example of $\sigma_\epsilon(\mathbf{T})$ generated with the Eigtool package [120], where $\mathbf{T}$ is the Tuesday Lake matrix [84].

by eigenvalues made the study of pseudospectra especially interesting in the 1990's [108, p.xiii-xiv]. Pseudospectra specifically are interesting because they are a way of organizing more information about how a matrix is non-normal than a single constant value can provide.

Trefethen is not the first researcher to discuss pseudospectra though. Varah (1979) [114], Demmel (1983, 1987) [35, 34], Wilkinson (1986) [119], Godunov et al. (1990) [53], and likely other mathematicians had published equivalent definitions with different names for the sets. However, after this 1991 re-introduction to pseudospectra, Trefethen continued to publish applications of ϵ-pseudospectral sets, and, in 2005, published *Spectra and Pseudospectra* with Mark Embree that collected much of the pseudospectral work of the intervening decade and a half.

4.3.1 ϵ-Pseudospectral Bounds

The ϵ-pseudospectrum may be used to calculate bounds on $\|f(\mathbf{A})\|_2$. For instance, ϵ-pseudospectral sets may be used to define the Kreiss constant $\mathscr{K}(\mathbf{A})$. The Kreiss constant is used in the Kreiss matrix theorem. One version of the Kreiss matrix theorem bounds $\|\mathbf{A}^k\|_2$ for k greater than or equal to zero and $\mathbf{A}$ where all eigenvalues are contained in $\mathcal{D}(0,1)$. Let the ϵ-pseudospectral radius be $w_\epsilon(\mathbf{A}) := \sup|\Lambda_\epsilon(\mathbf{A})|$, then the Kreiss constant of $\mathbf{A}$ with respect to the unit disk is

$$\mathscr{K}(\mathbf{A}) = \sup_{\epsilon}(w_\epsilon(\mathbf{A}) - 1)/\epsilon = \sup_{|z|>1}(|z| - 1)\|(z\mathbf{I} - \mathbf{A})^{-1}\|_2. \tag{4.4}$$

In this case, the Kriess matrix theorem states that

$$\mathscr{K}(\mathbf{A}) \leq \sup_{k\geq 0}\|\mathbf{A}^k\|_2 \leq en\mathscr{K}(\mathbf{A}). \tag{4.5}$$

Another version of the Kriess matrix theorem bounds $\|\exp(t\mathbf{A})\|_2$ for t greater than or equal to zero and $\mathbf{A}$ where all eigenvalues have negative real parts. Let the ϵ-pseudospectral abscissa be $\rho_\epsilon(\mathbf{A}) := \sup(\mathrm{Re}(\Lambda_\epsilon(\mathbf{A})))$, then the Kreiss constant of $\mathbf{A}$ with respect to the left-half plane is

$$\mathscr{K}(\mathbf{A}) := \sup_{\epsilon}\rho_\epsilon(\mathbf{A})/\epsilon = \sup_{\mathrm{Re}(z)>0}\mathrm{Re}(z)\|(z\mathbf{I} - \mathbf{A})^{-1}\|_2. \tag{4.6}$$

In this case, the Kriess matrix theorem states that

$$\mathscr{K}(\mathbf{A}) \leq \sup_{t\geq 0}\|\exp(t\mathbf{A})\|_2 \leq en\mathscr{K}(\mathbf{A}). \tag{4.7}$$

The Kriess matrix theorem can also be extended to bound any holomorphic function on a disk or half-plane containing all of the eigenvalues by using conformal maps [92].

We may also use pseudospectral sets to bound the norm of the Cauchy integral. The

Cauchy integral is

$$f(\mathbf{A}) = \frac{1}{2\pi i}\int_{\partial\Omega}(\zeta\mathbf{I}-\mathbf{A})^{-1}f(\zeta)\mathrm{d}\zeta$$
$$\Rightarrow \|f(\mathbf{A})\|_2 \leq \frac{1}{2\pi}\left(\int_{\partial\Omega}\|(\zeta\mathbf{I}-\mathbf{A})^{-1}\|_2|\mathrm{d}\zeta|\right)\|f\|_\Omega. \tag{4.8}$$

The norm $\|f\|_\Omega$ is equal to $\sup_{z\in\Omega}|f(z)|$. The integrand of equation (4.8) is the norm of the resolvent. Using (4.8) and taking Ω to be an ϵ-pseudospectrum set, we can bound $\|f(\mathbf{A})\|_2$.

Theorem 4.2: [108, §14]
Let $\Omega \subset \mathbb{C}$ and $\mathcal{L}(\partial\Omega)$ be the length of the boundary. Let $\Lambda_\epsilon(\mathbf{A}) \subseteq \Omega$ and the function f be analytic in Ω. Then,

$$\|f(\mathbf{A})\|_2 \leq \frac{\mathcal{L}(\partial\Omega)}{2\pi\epsilon}\sup_{z\in\Omega}|f(z)|. \tag{4.9}$$

Notice that one option for the set Ω is always $\Lambda_\epsilon(\mathbf{A})$.

Having prior knowledge about pseudospectral sets can simplify the calculation of some bounds on $\|f(\mathbf{A})\|_2$, including calculating the Kreiss matrix consant and the Cauchy integral. At their core, pseudospectral sets are a useful tool for measuring and visualizing the sensitivity of a matrix's eigenvalues.

4.4 Background on K-Spectral Sets

Spectral sets have been used in the field of functional analysis for a long time. In 1951, Von Neumann published foundational results relating a polynomial function of an operator to the polynomial's behavior on a subset of the complex plane [116]. Let p be a polynomial function, and $\mathbf{A}$ be a matrix with eigenvalues contained in the unit disk $\mathcal{D}$. Von Neumann proved that

$$\|p(\mathbf{A})\| \leq \sup_{z\in\mathcal{D}}|p(z)|. \tag{4.10}$$

A K-Spectral set is a generalization of equation (4.10).

Definition 4.4. *Let $\mathbf{A} \in \mathbb{C}^{N\times N}$ or a bounded linear operator on a complex Hilbert space*

$(H, \langle\cdot,\cdot\rangle, \|\cdot\|)$. *A closed set* $\Omega \subset \mathbb{C}$ *is a* K***-spectral set*** *for* $\mathbf{A}$ *if the eigenvalues of* $\mathbf{A}$ *are contained in* Ω*, and if, for all functions* f *analytic in* Ω*, where analytic functions in a region can be arbitrarily well-approximated by bounded rational functions on that same set, the following inequality holds:*

$$\|f(\mathbf{A})\| \leq K\|f\|_\Omega \tag{4.11}$$

where $\|\cdot\|$ *on the left denotes the norm in* H *and* $\|\cdot\|_\Omega$ *on the right denotes the* ∞*-norm over the set* Ω*.*

Definition 4.4 may be extended to all meromorphic functions bounded on Ω if we make an additional assumption about the analytic capacity of the inner boundary curves of the region Ω, since such functions can be uniformly approximated by rational functions [115]. The constraint that Ω contains the spectrum of $\mathbf{A}$ is necessary to ensure that $\|f(\mathbf{A})\|$ can be bounded by a finite value K, but otherwise Ω can take on many different shapes.

The infinity norm in equation (4.11) may also be written as

$$\|f(\mathbf{A})\| \leq K \sup_{z\in\Omega} |f(z)|,$$

which clarifies that in equation (4.10) Ω is the unit disk and K is one. After Von Neumann's foundational result, much of the immediately following research about what will become known as K-spectral sets was focused on finding a value of K for specific operators and/or specific sets. However in 2004, Crouzeix published the conjecture that for the numerical range of any finite linear operator, $K = 2$ [27]. This conjecture and the result proving $K = 11.08$ in the same paper are powerful, because they prove K is a value that is not unique for each matrix and does not depend on the dimension of the matrix. The conjecture is that the proven value of K can be decreased by an order of magnitude. Thus far, for the set $W(\mathbf{A})$, a value of $K = 2$ has only been proven for matrices meeting very specific conditions [27, 52, 25]. However, in 2017 $K = 1+\sqrt{2}$ was proven for $W(\mathbf{A})$ for every matrix, improving the bound from 11.08 to approximately 2.414 [30].

Another extension to K-spectral set theory is Crouzeix and Greenbaum's 2019 paper that bounds the value of K for the set $\Omega = W(\mathbf{A})$ with a single disk removed [29]. Our contributions extend the result of bounding K after removing a single disk from the numerical range to bounding K for removing any finite number of disks. We also numerically implement a calculation of K based on a theorem first published in [29] for any region Ω. In the rest of this chapter we review the results from Crouzeix and Grenbaum (2019). Our results are introduced in Chapter 5.

4.4.1 Definitions

Let f be a rational function bounded in a closed set Ω containing the spectrum of $\mathbf{A}$. When the arc length of a curve is approximated by connecting a finite number of points on the curve using straight line segments, the supremum of these lengths is the arc length L. A rectifiable curve is one with a finite total arc length. A connected curve is one where any two points on the curve can be reached by following a path belonging to the curve. Assume that the boundary of the closed set Ω is $\partial\Omega$ and that $\partial\Omega$ is rectifiable and has a finite number of connected components.

The Cauchy integral theorem states that

$$f(z) = \frac{1}{2\pi i}\int_{\partial\Omega} \frac{f(\zeta)}{\zeta - z}\, d\zeta, \text{ and } f(\mathbf{A}) = \frac{1}{2\pi i}\int_{\partial\Omega} (\zeta\mathbf{I} - \mathbf{A})^{-1} f(\zeta)\, d\zeta$$

for matrices. The above equations can be rewritten in parametric form by letting $0 = s_0 \leq s \leq L$ denote arc length, and letting $\zeta(s)$ be the location in $\partial\Omega$ that is a distance along the boundary of s traveled from $\zeta_0 = \zeta(s_0)$ (excluding jumps between disconnected portions of $\partial\Omega$) while traveling in a counter-clockwise direction. Then the parametric forms of $f(\cdot)$ are

$$\begin{aligned} f(z) =& \frac{1}{2\pi i}\int_{\{s:\zeta(s)\in\partial\Omega\}} \frac{f(\zeta(s))}{\zeta(s) - z}\zeta'(s)\, ds, \text{ and} \\ f(\mathbf{A}) =& \frac{1}{2\pi i}\int_{\{s:\zeta(s)\in\partial\Omega\}} (\zeta(s)\mathbf{I} - \mathbf{A})^{-1} f(\zeta(s))\zeta'(s)\, ds. \end{aligned}$$

If $\partial\Omega$ has multiple connected components, then $\zeta(s)$ is assumed to entirely traverse one connected component before continuing on the next one.

We also use the Cauchy transform of the complex conjugate $\bar{f}$:

$$g(z) := C(\overline{f}, z) := \frac{1}{2\pi i}\int_{\{s:\zeta(s)\in\partial\Omega\}} \frac{\overline{f(\zeta(s))}}{\zeta(s)-z}\zeta'(s)\,ds, \text{ and}$$
$$g(\mathbf{A}) := \frac{1}{2\pi i}\int_{\{s:\zeta(s)\in\partial\Omega\}} (\zeta(s)\mathbf{I}-\mathbf{A})^{-1}\overline{f(\zeta(s))}\zeta'(s)\,ds.$$

Finally, we define the double layer potential kernel,

$$\mu(\zeta(s), z) := \frac{1}{\pi}\frac{d}{ds}(\arg(\zeta(s)-z)) = \frac{1}{2\pi i}\left(\frac{\zeta'(s)}{\zeta(s)-z} - \frac{\overline{\zeta'(s)}}{\overline{\zeta(s)}-\bar{z}}\right), \tag{4.12}$$

$$\mu(\zeta(s), \mathbf{A}) = \frac{1}{2\pi i}\left((\zeta(s)\mathbf{I}-\mathbf{A})^{-1}\zeta'(s) - (\overline{\zeta(s)}\mathbf{I}-\mathbf{A}^*)^{-1}\overline{\zeta'(s)}\right). \tag{4.13}$$

4.4.2 Analytic Bound on K when a Single Disk is Removed from the Numerical Range

This section contains some of the main results from Crouzeix and Greenbaum (2019)[29].

To calculate the value of K for a convex set containing the numerical range except for a single disk removed, we first calculate the value of two constants, c_1 and c_2.

First, define

$$c_1 := \sup\{\sup_{z\in\Omega} |C(\bar{f}, z)| : f \text{ a rational function}, \|f\|_\Omega \le 1\}. \tag{4.14}$$

It is shown in [29, Lemma 1] that c_1 satisfies

$$c_1 \le \operatorname*{supess}_{\zeta_0\in\partial\Omega} \int_{\{s:\zeta(s)\in\partial\Omega\}} |\mu(\zeta(s), \zeta_0)|\,ds. \tag{4.15}$$

To define c_2, we first define the transform of f,

$$S(f, z) := f(z) + \overline{g(z)} = \int_{\{s:\zeta(s)\in\partial\Omega\}} f(\zeta(s))\mu(\zeta(s), z)\,ds,$$

$$S(f, \mathbf{A}) := f(\mathbf{A}) + g(\mathbf{A})^* = \int_{\{s:\zeta(s)\in\partial\Omega\}} f(\zeta(s))\mu(\zeta(s), \mathbf{A})\, ds.$$

Note that $S(1, \mathbf{A}) = 2\mathbf{I}$ since

$$\begin{aligned}\int_{\partial\omega} \mu(\zeta(s), \mathbf{A})\, ds =& \frac{1}{2\pi i} \int_{\{s:\zeta(s)\in\partial\Omega\}} (\zeta(s)\mathbf{I} - \mathbf{A})^{-1}\zeta'(s)\, ds \\ &+ \left(\frac{1}{2\pi i} \int_{\{s:\zeta(s)\in\partial\Omega\}} (\zeta(s)\mathbf{I} - \mathbf{A})^{-1}\zeta'(s)\, ds\right)^* \\ =& \mathbf{I} + \mathbf{I}^* = 2\mathbf{I}.\end{aligned}$$

This function S is defined using the Cauchy transform of the complex conjugate of f and the Cauchy integral of f, which can be summarized using the double layer potential kernel μ. These definitions are all found in Section 4.4.1. Now we define

$$c_2 := \frac{1}{2} \sup\{\|S(f, \mathbf{A})\| : f \text{ a rational function}, \|f\|_\Omega \leq 1\}. \tag{4.16}$$

The following is (a part of) the main theorem of Crouzeix and Greenbaum (2019) [29, Theorem 2]:

Theorem 4.3:
Let $\mathbf{A}$ be a square matrix. Let c_1 and c_2 be defined by (4.14) and (4.16) respectively. Then for any $\Omega \subset \mathbb{C}$ containing the spectrum of $\mathbf{A}$, Ω is a K-spectral set for $\mathbf{A}$, where

$$K = c_2 + \sqrt{c_2^2 + c_1}.$$

We are interested in calculating the value of K for the numerical range of a matrix with a single disk removed. The numerical range is a convex set containing the eigenvalues of $\mathbf{A}$, which means it contains the eigenvalues of the matrix. We use equations (4.15) and (4.12) to bound c_1. First, fix $\zeta_0 \in \partial\Omega$ and let $\zeta(s)$ move around a curve Γ_j that is a connected curve of $\partial\Omega$. The curve Γ_j may be all or part of $\partial\Omega$, and may be all or part of a single

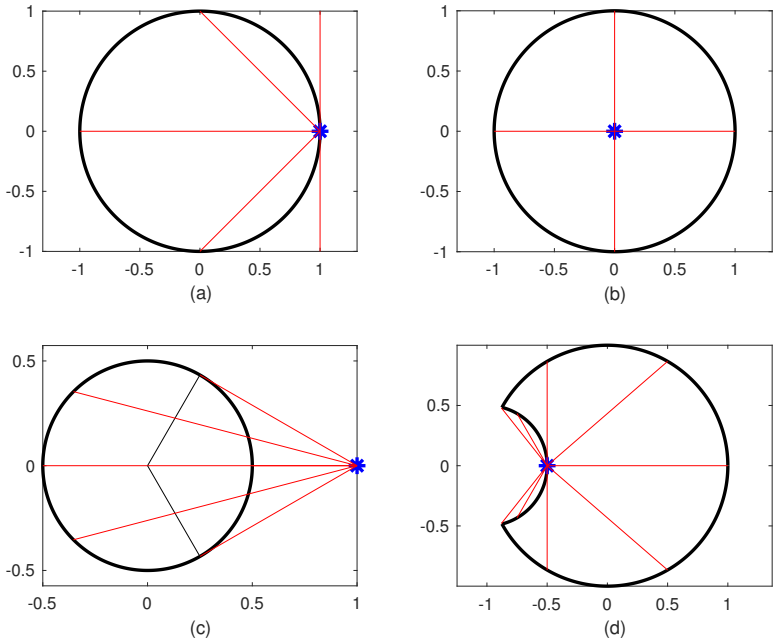


Figure 4.3: Various boundary configurations. The blue asterisk represents ζ_0, and the red lines show how the angle of the vector $\zeta(s) - \zeta_0$ changes as $\zeta(s)$ traverses the boundary curve.

connected component of $\partial\Omega$. Then from equation (4.12), $\int_{s:\zeta(s)\in\Gamma_j} |\mu(\zeta(s),\zeta_0)|\,ds$ is equal to $\frac{1}{\pi}$ times the total variation in the argument of $\zeta(s)-\zeta_0$. For example, if $\partial\Omega$ is the boundary of a convex set like a disk, as shown in Figure 4.3(a), then the argument of $\zeta(s)-\zeta_0$ changes by π as $\zeta(s)$ traverses the curve $\partial\Omega$ regardless of where ζ_0 is located on $\partial\Omega$. Since the total variation is π, then from equation (4.12) we see that $\int_{s:\zeta(s)\in\partial\Omega} |\mu(\zeta(s),\zeta_0)|\,ds = 1$. When there are multiple connected components defining $\partial\Omega$, such as when Ω is an annulus, then it is possible ζ_0 lies inside a convex set or on the outside of a convex set. If ζ_0 lies inside a circle or the boundary curve of a convex set such as in Figure 4.3(b), then the total variation of the angle from ζ_0 is 2π and the integral of $|\mu(\zeta(s),\zeta_0)|$ over that piece of the boundary is 2. If ζ_0 lies outside a circle of radius r such as in Figure 4.3(c), then, if q is the distance from ζ_0 to the center of the circle, the argument of $\zeta(s)-\zeta_0$ goes from its initial value, say, 0 to $\arcsin(r/q)$ to 0, to $-\arcsin(r/q)$, and back to 0, for a total change of $4\arcsin(r/q)$.

To obtain upper bounds on c_2, we first note that if $\mu(\zeta(s), \mathbf{A})$ is positive semidefinite (PSD) along Γ_j for $s \in [s_{min}, s_{max}]$, then

$$\left\| \int_{s_{min}}^{s_{max}} f(\zeta(s))\mu(\zeta(s), \mathbf{A})\, ds \right\| \leq \max_{s\in[s_{min},s_{max}]} |f(\zeta(s))| \ \left\| \int_{s_{min}}^{s_{max}} \mu(\zeta(s), \mathbf{A})\, dz \right\|. \tag{4.17}$$

A proof can be obtained by noting that

$$\left\| \int_{s_{min}}^{s_{max}} f(\zeta(s))\mu(\zeta(s), \mathbf{A})\, ds \right\| = \sup_{\|x\|=\|y\|=1} \left| \int_{s_{min}}^{s_{max}} f(\zeta(s)) \left\langle \mu(\zeta(s), \mathbf{A})y, x \right\rangle \, ds \right|,$$

and following the arguments in [30, Lemma 2.3]. Thus if $\mu(\zeta, \mathbf{A})$ is PSD for all $\zeta \in \partial\Omega$, then $c_2 \leq 1$, since for any rational function f with $\|f\|_\Omega \leq 1$,

$$\|S(f, \mathbf{A})\| \leq \left\| \int_{\{s:\zeta(s)\in\partial\Omega\}} \mu(\zeta(s), \mathbf{A})\, ds \right\| = \|2\mathbf{I}\| = 2,$$

and from equation (4.16) c_2 is bounded by half this value. For convex Ω with PSD $\mu(\zeta(s), \mathbf{A})$ for all of $\partial\Omega$ the value of c_1 and c_2 is bounded by one, and Theorem 4.3 states that $K = 1+\sqrt{2}$, which matches the value of K in the Crouzeix-Palencia result [30]. Below, Theorem 4.4 will show that $\mu(\zeta(s), \mathbf{A})$ is PSD for all of $\partial W(\mathbf{A})$, which is equivalent to the Crouzeix-Palencia result [30].

For any subset of $\partial\Omega$ where $\mu(\zeta(s), \mathbf{A})$ is not PSD, we will add a multiple of the identity to $\mu(\zeta(s), \mathbf{A})$ to obtain a PSD operator for all values of s. The multiple of the identity added to $\mu(\zeta(s), \mathbf{A})$ must be at least as large as the minimum eigenvalue of $\mu(\zeta(s), \mathbf{A})$ to have a PSD operator. This means we need bounds on the minimum value in the spectrum of $\mu(\zeta(s), \mathbf{A})$:

$$\lambda_{min}(\mu(\zeta(s), \mathbf{A})) := \min\{\lambda : \lambda \in \mathrm{Sp}(\mu(\zeta(s), \mathbf{A}))\}. \tag{4.18}$$

Let $\zeta_0 = \zeta(s_0)$ denote a point on $\partial\Omega$ where the unit tangent $\zeta_0' := \frac{d\zeta}{ds}\Big|_{s_0}$ exists. We write $\mu(\zeta_0, \zeta_0', \mathbf{A})$ when we fix a point ζ_0 since in this case $\mu(\zeta(s), \mathbf{A})$ depends on $\zeta'(s)$. Note that the half-plane $\Pi_0 := \{z \in \mathbb{C} : \mathrm{Im}(\zeta_0'(\overline{\zeta_0} - \bar{z})) \geq 0\}$ has the same outward normal as Ω

at ζ_0. Recall, the numerical range is defined in Definition 4.1, and the numerical radius is $w(\mathbf{A}) = \sup|W(\mathbf{A})|$. The following theorem is from [29, Lemmas 5, 7, and 8].

Theorem 4.4:

Let $\mathbf{A}$ be a square matrix, $W(\mathbf{A})$ be the numerical range of $\mathbf{A}$, $\Pi_0 := \{z \in \mathbb{C} : \mathrm{Im}(\zeta_0'(\overline{\zeta_0} - \bar{z})) \geq 0\}$, $\lambda_{\min}$ be defined by (4.18), and $\mu(\zeta_0, \zeta_0', \mathbf{A})$ be defined by (4.13) at the point s_0. If $W(\mathbf{A}) \subset \Pi_0$, then $\lambda_{min}(\mu(\zeta_0, \zeta_0', \mathbf{A})) \geq 0$, with equality if $\zeta_0 \in \partial W(\mathbf{A})$. If, for some $\xi \in \mathbb{C}\backslash\mathrm{Sp}(\mathbf{A})$, $\zeta_0 - \xi = ir_1\zeta_0'$, where $r_1 \leq 1/\|(\xi\mathbf{I} - \mathbf{A})^{-1}\|$, then $\lambda_{min}(\mu(\zeta_0, \zeta_0', \mathbf{A})) \geq -\frac{1}{2\pi r_1}$. If $\zeta_0 - \xi = ir_2\zeta_0'$, where $r_2 \leq 1/w((\xi\mathbf{I} - \mathbf{A})^{-1})$, then $\lambda_{min}(\mu(\zeta_0, \zeta_0', \mathbf{A})) \geq -\frac{1}{\pi r_2}$.

For a disk about a point ξ of radius r, the assumption $\zeta_0 - \xi = ir\zeta_0'$ in the theorem means that $\partial\Omega$ and the boundary of the disk are tangent at ζ_0 and the outward normal to Ω, ζ_0'/i, is the same as the inward normal to the disk. Further, the interior of the disks $\{z \in \mathbb{C} : |z - \xi| < 1/\|(\xi\mathbf{I} - \mathbf{A})^{-1}\|\}$ and $\{z \in \mathbb{C} : |z - \xi| < 1/w((\xi\mathbf{I} - \mathbf{A})^{-1})\}$ alluded to in the theorem contain no points in the spectrum of $\mathbf{A}$. The spectrum of $\mathbf{A}$ is not contained in the disks since $\|(\xi\mathbf{I} - \mathbf{A})^{-1}\| \geq w((\xi\mathbf{I} - \mathbf{A})^{-1}) \geq |(\lambda - \xi)^{-1}|$ for all $\lambda \in \mathrm{Sp}(\mathbf{A})$, which means that the inverses of these quantities, which are the radii of the disks, are less than or equal to $|\lambda - \xi|$.

Theorems 4.3 and 4.4 can be used together to obtain K values for certain types of sets, such as the numerical range with a circular hole or cutout. In the next subsection, we include an example from [29] of a K-spectral set equal to the numerical range of a matrix with a single disk removed, where the disk is defined using Theorem 4.4.

4.4.3 Example from [29]

Let Ω_0 be a convex domain containing the numerical range of the $\mathbf{A}$, $W(\mathbf{A})$. Let $\mathcal{D}(\xi, r)$ be the disk about a point $\xi \in \mathbb{C}\backslash\mathrm{Sp}(\mathbf{A})$ of radius r, where $r \leq 1/w((\xi\mathbf{I} - \mathbf{A})^{-1})$. Then it is shown in [29] that $\Omega = \Omega_0\backslash\mathcal{D}(\xi, r)$ is a $(3 + 2\sqrt{3})$-spectral set for $\mathbf{A}$. For K to equal $(3 + 2\sqrt{3})$, we assume that either $\partial\mathcal{D}(\xi, r) \subset \Omega_0$ or the number of intersection points of $\partial\Omega_0$ and $\partial\mathcal{D}(\xi, r)$ is finite.

To bound c_1 when removing a disk from a convex set, suppose first that $\partial\mathcal{D}(\xi, r) \subset \Omega_0$. If $\zeta_0 \in \partial\Omega_0$, then as $\zeta(s)$ traverses $\partial\Omega_0$, the argument of $\zeta(s) - \zeta_0$ changes by π, as illustrated in Figure 4.3(a). As $\zeta(s)$ traverses $\partial\mathcal{D}(\xi, r)$, the argument of $\zeta(s) - \zeta_0$ changes by $4 \arcsin(r/|\xi - \zeta_0|) < 2\pi$, as illustrated in Figure 4.3(c). Thus in this case,

$$\int_{\{s:\zeta(s)\in\partial\Omega_0\}} |\mu(\zeta(s), \zeta_0)|\, ds = 1, \quad \int_{\{s:\zeta(s)\in\partial\mathcal{D}(\xi,r)\}} |\mu(\zeta(s), \zeta_0)|\, ds < 2.$$

To simplify notation, throughout the rest of dissertation we will write $\int_{\partial\Omega_j} \dots ds$ in place of $\int_{\{s:\zeta(s)\in\partial\Omega_j\}} \dots ds$.

Now suppose $\zeta_0 \in \partial\mathcal{D}(\xi, r)$. Then as illustrated in Figure 4.3(b), as $\zeta(s)$ traverses $\partial\Omega_0$, the argument of $\zeta(s) - \zeta_0$ changes by 2π. As illustrated in Figure 4.3(a), while $\zeta(s)$ traverses $\partial\mathcal{D}(\xi, r)$, the argument of $\zeta(s) - \zeta_0$ changes by π. Thus in this case

$$\int_{\partial\Omega_0} |\mu(\zeta(s), \zeta_0)|\, ds = 2, \quad \int_{\partial\mathcal{D}(\xi,r)} |\mu(\zeta(s), \zeta_0)|\, ds = 1.$$

If, instead, the disk $\mathcal{D}(\xi, r)$ intersects $\partial\Omega_0$ as in Figure 4.3(d), then it is clear that the total variation in the argument of $\zeta(s) - \zeta_0$ as $\zeta(s)$ traverses $\partial\Omega$ is smaller than 3π and thus c_1 is bounded by 3. Thus when Ω is a convex set with a single disk removed, it follows that for ζ_0 anywhere on the boundary of Ω, the change in argument of $\zeta(s) - \zeta_0$ as $\zeta(s)$ traverses $\partial\Omega$ is at most 3π and $c_1 \leq 3$.

To bound c_2, let $\Gamma_0 = \partial\Omega_0 \backslash \mathrm{cl}(\mathcal{D}(\xi, r))$ and let $\Gamma_1 = \partial\mathcal{D}(\xi, r) \cap \mathrm{cl}(\Omega_0)$ so that $\partial\Omega = \Gamma_0 \cup \Gamma_1$. Let f be a function such that $\|f\|_\Omega \leq 1$. Then we write $S(f, \mathbf{A}) = S_0 + S_1 + S_2$, where

$$\begin{aligned}
S_0 &= \int_{\Gamma_0} f(\zeta(s))\mu(\zeta(s), \mathbf{A})\, ds, \\
S_1 &= \int_{\Gamma_1} f(\zeta(s)) \left(\mu(\zeta(s), \mathbf{A}) + \frac{1}{\pi r}\mathbf{I}\right) ds, \\
S_2 &= -\frac{1}{\pi r} \int_{\Gamma_1} f(\zeta(s))\mathbf{I}\, ds.
\end{aligned}$$

It follows from Theorem 4.4 that for $\zeta \in \partial\Omega_0$, $\mu(\zeta, \mathbf{A})$ is PSD. Since adding PSD operators

to a PSD operator does not decrease the norm, we can extend the integral over Γ_0 to an integral over the entire boundary $\partial\Omega_0$ to obtain:

$$\|S_0\| \leq \left\| \int_{\partial\Omega_0} \mu(\zeta(s), \mathbf{A})\, ds \right\| = \|2\mathbf{I}\| = 2.$$

If $\zeta \in \partial\mathcal{D}(\xi, r)$, since $r \leq 1/w((\xi\mathbf{I} - \mathbf{A})^{-1})$, Theorem 4.4 shows that $\mu(\zeta, \mathbf{A}) + \frac{1}{\pi r}\mathbf{I}$ is PSD, and hence

$$\begin{aligned}
\|S_1\| \leq & \left\| \int_{\Gamma_1} \left(\mu(\zeta(s), \mathbf{A}) + \frac{1}{\pi r}\mathbf{I} \right) ds \right\| \\
\leq & \left\| \int_{\partial\mathcal{D}(\xi,r)} \left(\mu(\zeta(s), \mathbf{A}) + \frac{1}{\pi r}\mathbf{I} \right) ds \right\| \\
= & \frac{1}{\pi r} \int_{\partial\mathcal{D}(\xi,r)} ds \\
= & 2.
\end{aligned}$$

Here we used that the spectrum of $\mathbf{A}$ lies outside $\mathcal{D}(\xi, r)$ so that $\int_{\partial\mathcal{D}(\xi,r)} \mu(\zeta(s), \mathbf{A})\, ds = 0$. Further, $\|S_2\| \leq 2$ since the length of Γ_1 is less than or equal to the length of $\partial\mathcal{D}(\xi, r)$, which is $2\pi r$. Thus $\|S(f, \mathbf{A})\| = \|S_0 + S_1 + S_2\| \leq 6$ and $c_2 \leq 3$. Applying Theorem 4.3 with $c_1 = c_2 = 3$ yields the result from [29] that Ω is a $(3 + 2\sqrt{3})$-spectral set for $\mathbf{A}$.

In the next chapter, we extend this example in several ways, indicate how Theorem 4.3 can be used directly to determine a K value for any set Ω containing the spectrum of $\mathbf{A}$, and describe how to numerically estimate an upper bound of K for any K-spectral set Ω.

Chapter 5

K-SPECTRAL SETS

In this chapter, our work extends the arguments in Section 4.4.2. Let Ω be a K-spectral set of $\mathbf{A}$ such that for any function f analytic in Ω (i.e. any analytic function may be arbitrarily well-approximated on Ω by a rational function without poles in the set Ω)

$$\|f(\mathbf{A})\|_2 \leq K \sup_{z\in\Omega} |f(z)|.$$

In this chapter, we obtain a bound on K for convex sets containing the numerical range with multiple disks removed, and we show how to directly calculate K for any region Ω containing the spectrum of $\mathbf{A}$. One motivation for removing a disk from the numerical range comes from the study of iterative method convergence. For example, when GMRES is used to solve $\mathbf{Ax} = \mathbf{b}$, then we are using GMRES to compute $f(\mathbf{A})\mathbf{b}$ where $f(z) = 1/z$, and the function f has a pole at zero. This means that if the numerical range contains the origin, then for $\Omega = W(\mathbf{A})$ the $\sup_{z\in\Omega} |f(z)|$ is infinite. However, there are many cases where $W(\mathbf{A})$ contains the origin and GMRES converges without issue to a solution. By defining the K-spectral set Ω to be the numerical range without a disk centered at 0, then $\sup_{z\in\Omega} |f(z)|$ is finite. Our work removing multiple disks allows us to bound iterative methods dealing with rational functions of $\mathbf{A}$ with more than one pole in the numerical range, as discussed in Chapter 3.

Another goal for bounding K using the techniques in the proofs of Theorems 4.3 and 4.4 was to generate smaller bounds on K than one would get by applying the Cauchy integral formula directly to $\partial\Omega$ and replacing the norm of the integral by the integral of the norm. We show that while the Cauchy integral formula in equation (4.8) leads to a smaller bound in some cases, it cannot be smaller by an amount that exceeds a factor of 4 plus a small

amount. However, the reverse is not true, and the method used to bound K using Theorem 4.3 may give a much smaller bound than the bound on K using (4.8).

In Section 5.1 we tighten the analytic bounds for three specific cases given in [29], two of which involve removing at most a half disk from a convex region containing $W(\mathbf{A})$. In Sections 5.2 and 5.3 we extend the results from [29] to any spectral set in the complex plane and show that the method based on the Cauchy Integral formula can only improve on the results from [29] by a factor of $4+\epsilon$, whereas there is no bound on the improvement that the methodology in [29] can make in bounding K. In Section 5.4, we give a partial explanation as to when and why one method may do better than the other for different matrices. Then in Section 5.5 we include examples for the reader illustrating different types of spectral sets and different use cases for removing disks or portions of the numerical range. In some of these examples using the method based on the Cauchy integral to bound K will yield a smaller value, and in others using the methodology that we extended based on Theorem 4.3 will yield the smaller value.

5.1 Bounds for a Half Disk Removed

First, we define a tighter bound for the example in Figure 4.3(d) where at most a half-disk is removed from the numerical range. Let $\Omega = \Omega_0 \backslash \mathcal{D}(\xi, r)$ where Ω_0 is a convex set containing the numerical range and $\mathcal{D}(\xi, r)$ is a disk with center ξ and radius $r \leq 1/w\left((\xi\mathbf{I} - \mathbf{A})^{-1}\right)$ as in Section 4.4.3. Assume the intersection of Ω_0 and $\mathcal{D}(\xi, r)$ is at most a half-disk. Let $\Gamma_0 = \partial\Omega_0 \backslash \text{cl}(\mathcal{D}(\xi, r))$ be the portion of $\partial\Omega$ containing the boundary of the numerical range, and let $\Gamma_1 = \partial\mathcal{D}(\xi, r) \cap \Omega_0$ be the portion of $\partial\Omega$ containing the boundary of the disk removed. Then $\partial\Omega = \Gamma_0 \cup \Gamma_1$. Let s be the parameterization of ζ so that $\zeta(s)$ is a continuous function of s such that for increasing values of s, $\zeta(s)$ travels counter-clockwise along Γ. Let $\zeta(s) = \zeta(s_0) = \zeta_0$ and $\zeta_0' = \left.\frac{d\zeta}{ds}\right|_{s_0}$. Then the greatest variation in the argument of $\zeta_0 - \zeta(s)$ occurs when ζ_0 is located at the asterisk in Figure 4.3(d), which is half-way along the curve Γ_1. When ζ_0 is located at that asterisk, then the total variation of the argument of $\zeta(s) - \zeta_0$ can change by up to $\pi/2$ as $\zeta(s)$ traverses Γ_1. The total variation of

the argument of $\zeta(s) - \zeta_0$ changes by the same amount as $\zeta(s)$ moves along Γ_0 to the point where the argument of $\zeta(s) - \zeta_0$ matches ζ_0' or $-\zeta_0'$, with a change of π in between both those points. The total change could therefore be as large as 2π. It follows that in this case, for any ζ_0 on $\partial\Omega$,

$$\int_{\partial\Omega_0} |\mu(\zeta(s), \zeta_0)|\, ds \leq 2,$$

and therefore $c_1 \leq 2$ when at most a half-disk is removed from Ω_0. Using the same definitions of S_0, S_1, and S_2 as in Section 4.4.3, we now observe that the length of Γ_1 is at most πr instead of $2\pi r$, so that $\|S_2\|_2 \leq 1$, leading to the estimate $\|S(f, \mathbf{A})\|_2 \leq 5$ and $c_2 \leq 5/2$. Using these values of c_1 and c_2 in Theorem 4.3 leads to the result that Ω is a $(2.5 + \sqrt{8.25})$-spectral set of $\mathbf{A}$.

Next we drop the assumption that the disk removed is at most a half-disk. The examples so far have assumed that $r \leq 1/w\left((\xi\mathbf{I} - \mathbf{A})^{-1}\right)$. However, if the radius r of the disk removed from Ω_0 satisfies $r \leq 1/\|(\xi\mathbf{I} - \mathbf{A})^{-1}\|_2$, then from Theorem 4.4 it follows that $\lambda_{min}(\mu(\zeta_0, \mathbf{A})) \geq -\frac{1}{2\pi r}$. In this case, we replace $S(f, \mathbf{A}) = S_0 + S_1 + S_2$ with $S(f, \mathbf{A}) = S_0 + \tilde{S}_1 + \tilde{S}_2$, where

$$\begin{aligned}
\tilde{S}_1 &= \int_{\Gamma_1} f(\zeta(s)) \left(\mu(\zeta(s), \mathbf{A}) + \frac{1}{2\pi r}\mathbf{I}\right) ds, \text{ and} \\
\tilde{S}_2 &= -\frac{1}{2\pi r} \int_{\Gamma_1} f(\zeta(s))\mathbf{I}\, ds.
\end{aligned}$$

Now

$$\|\tilde{S}_1\|_2 \leq \left\| \int_{\Gamma_1} \left(\mu(\zeta(s), \mathbf{A}) + \frac{1}{2\pi r}\mathbf{I}\right) ds \right\|_2 \leq \frac{1}{2\pi r} \int_{\partial\mathcal{D}(\xi, r)} ds = 1,$$

and $\|\tilde{S}_2\|_2 \leq 1$. With $c_1 = 3$ and $c_2 = 2$, it follows from Theorem 4.3 that Ω is a $(2 + \sqrt{7})$-spectral set. Now suppose the intersection of Ω_0 with this disk $\mathcal{D}(\xi, r)$ with $r < 1/\|(\xi\mathbf{I} - \mathbf{A})^{-1}\|_2$ is at most a half-disk, like in the first example of this section and in Figure 4.3(d). Then $c_1 = 2$ and $\|\tilde{S}_2\|_2 \leq 1/2$, so we take $c_2 = 7/4$ and it follows from Theorem 4.3 that this is a 4-spectral set for $\mathbf{A}$.

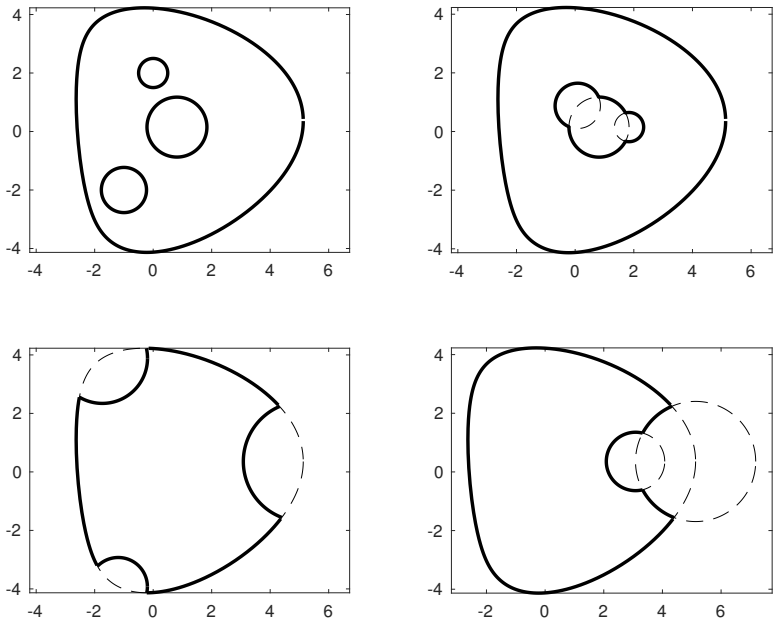


Figure 5.1: Regions with multiple holes or cutouts.

5.2 Removing More Disks

The techniques of Section 4.4.2 can also be used to bound K when a finite number of disks are removed from $\Omega_0 \supset W(\mathbf{A})$. Examples of the variety of geometries available when multiple disks are removed are shown in Figure 5.1.

Corollary 5.1:

Let $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\Omega_0 \subset \mathbb{C}$ and convex, and $W(\mathbf{A})$ denote the numerical range of $\mathbf{A}$. Suppose $\Omega_0 \supset W(\mathbf{A})$ and Ω is obtained from Ω_0 by removing m disks centered at points $\xi_1, \ldots, \xi_m \notin Sp(\mathbf{A})$, with the radius r_j of disk j bounded by either $1/\|(\xi_j\mathbf{I}-\mathbf{A})^{-1}\|_2$ or $1/w((\xi_j\mathbf{I}-\mathbf{A})^{-1})$. Set $p_j = 1$ if $r_j \leq 1/\|(\xi_j\mathbf{I}-\mathbf{A})^{-1}\|_2$ and $p_j = 2$ if $r_j \leq 1/w((\xi_j\mathbf{I}-\mathbf{A})^{-1})$ and $r_j > 1/\|(\xi_j\mathbf{I}-\mathbf{A})^{-1}\|_2$. Then Ω is a K-spectral set for $\mathbf{A}$ with

$$K \leq \left(1+\sum_{j=1}^{m} p_j\right) + \sqrt{\left(1+\sum_{j=1}^{m} p_j\right)^2 + 2m + 1}. \tag{5.1}$$

Proof. Consider first the simplest case, where the disks $\mathcal{D}_1(\xi_1, r_1), \ldots, \mathcal{D}_m(\xi_m, r_m)$ do not overlap and lie entirely inside Ω_0 like in the top left of Figure 5.1. For $\zeta_0 \in \partial\Omega_0$, the total variation in $\arg(\zeta(s) - \zeta_0)$ becomes

$$\pi + 4\sum_{j=1}^{m} \arcsin\left(\frac{1}{r_j|\zeta_0 - \xi_j|}\right) \leq \pi + 2m\pi.$$

If ζ_0 lies on $\partial\mathcal{D}_i$, then the change in $\arg(\zeta(s) - \zeta_0)$ is 2π as $\zeta(s)$ traverses $\partial\Omega_0$ and π as $\zeta(s)$ traverses $\partial\mathcal{D}_i$. The total change is

$$3\pi + 4\sum_{\substack{j=1 \\ j\neq i}}^{m} \arcsin\left(\frac{1}{r_j|\zeta_0 - \xi_j|}\right) \leq 3\pi + 2(m-1)\pi.$$

In either case, the total variation of $\arg(\zeta(s) - \zeta_0)$ is at most $(2m+1)\pi$, so that $c_1 \leq 2m+1$.

To bound c_2, we write $S(f, \mathbf{A}) = S_0 + \sum_{j=1}^{m} S_j + \sum_{j=1}^{m} S_{m+j}$, where

$$\begin{aligned} S_0 &= \int_{\partial\Omega_0} f(\zeta(s))\mu(\zeta(s), \mathbf{A})\, ds, \\ S_j &= \int_{\partial\mathcal{D}_j} f(\zeta(s)) \left(\mu(\zeta(s), \mathbf{A})) + \frac{p_j}{2\pi r_j}\mathbf{I}\right) ds, \\ S_{m+j} &= -\frac{p_j}{2\pi r_j} \int_{\partial\mathcal{D}_j} f(\zeta(s))\mathbf{I}\, ds, \quad j = 1, \ldots, m. \end{aligned}$$

Then

$$\|S_0\|_2 \leq 2, \quad \|S_j\|_2 \leq p_j, \quad \|S_{m+j}\|_2 \leq p_j, \quad j = 1, \ldots, m.$$

It follows that

$$\|S(f, \mathbf{A})\|_2 \leq 2 + 2\sum_{j=1}^{m} p_j,$$

and $c_2 \leq 1 + \sum_{j=1}^{m} p_j$. We obtain the K value in equation (5.1) by applying Theorem 4.3 with $c_1 = 2m + 1$ and $c_2 = 1 + \sum_{j=1}^{m} p_j$. This upper bound holds for other configurations as well, where c_1 and/or c_2 may be smaller because disks overlap or only partially intersect with Ω_0. □

Note that when the disks in Corollary 5.1 overlap or only partially intersect with Ω_0, better bounds on K may be attainable by considering each geometry individually.

5.3 Other K-Spectral Sets

In the previous section, we made use of Theorem 4.4 to derive Corollary 5.1 and values of K that are independent of the matrix $\mathbf{A}$ for special types of regions Ω (that *do* depend on $\mathbf{A}$). For a given matrix $\mathbf{A}$ and set Ω containing the eigenvalues of $\mathbf{A}$, one can use Theorem 4.3 directly to derive K values. These K values will depend on both $\mathbf{A}$ and Ω, unlike when using Corollary 5.1 to remove one or more disks from $W(\mathbf{A})$. The value of K and/or the region Ω will usually need to be computed numerically. For example, when using Corollary 5.1, even if one has an analytic formula for $W(\mathbf{A})$ or a convex set Ω_0 containing $W(\mathbf{A})$, one is unlikely to have an analytic formula for $\|(\xi\mathbf{I}-\mathbf{A})^{-1}\|_2$ or $w((\xi\mathbf{I}-\mathbf{A})^{-1})$. To numerically bound K using methods based on Theorem 4.3, we first need to bound c_1 and c_2. A bound on the parameter c_1 depends only on the geometry of Ω, while a bound on the parameter c_2 can be computed using the values of $\lambda_{min}(\mu(\zeta(s),\mathbf{A}))$, which are usually numerically estimated.

Examples of regions Ω that might be of interest include the intersection of $W(\mathbf{A})$ with the left half-plane and the intersection of $W(\mathbf{A})$ with the unit disk. Choosing Ω equal to the intersection of $W(\mathbf{A})$ with the left half-plane is interesting when the spectrum of $\mathbf{A}$ lies in the left half-plane but the numerical range extends into the right half-plane. In this case if Ω is a K-spectral set for $\mathbf{A}$, then K is an upper bound on the amount that the norm of the solution to the continuous time dynamical system $y'(t) = \mathbf{A}y(t)$, $t > 0$ can grow over its initial value before eventually decaying to zero. Choosing Ω equal to the intersection of $W(\mathbf{A})$ with the unit disk is interesting when the spectrum of $\mathbf{A}$ lies within the unit disk but the numerical range extends beyond the unit disk. In this second case, if Ω is a K-spectral set for $\mathbf{A}$, then K is an upper bound on the amount by which the norms of powers of $\mathbf{A}$, $\|\mathbf{A}^j\|_2$, $j = 0, 1, \ldots$ can grow before decaying to zero.

In either of these cases, when $\Omega = W(\mathbf{A}) \cap$ (left half-plane) or $\Omega = W(\mathbf{A}) \cap \mathcal{D}(0,1)$,

Ω is convex so $c_1 = 1$. To bound c_2, let Γ_0 denote the portion of $\partial\Omega$ that belongs to $\partial W(\mathbf{A})$, and let Γ_1 denote the portion of $\partial\Omega$ that belongs to the line segment or circular arc resulting from the intersection of $W(\mathbf{A})$ with the imaginary axis or the unit circle. Then $\partial\Omega = \Gamma_0 \cup \Gamma_1$. For rational f bounded in the set Ω and continuous in $\partial\Omega$ with $\|f\|_\Omega \leq 1$, define

$$\begin{aligned} S_0 &= \int_{\Gamma_0} f(\zeta(s))\mu(\zeta(s), \mathbf{A})\, ds, \\ S_1 &= \int_{\Gamma_1} f(\zeta(s))(\mu(\zeta(s), \mathbf{A}) + \gamma(s)\mathbf{I})\, ds, \\ S_2 &= -\int_{\Gamma_1} f(\zeta(s))\gamma(s)\mathbf{I}\, ds, \end{aligned}$$

where $\gamma(s) \geq \max\{-\lambda_{min}(\mu(\zeta(s), \mathbf{A})),\ 0\}$. Since $\mu(\zeta(s), \mathbf{A})$ is positive semi-definite (PSD) for $\zeta(s) \in \partial W(\mathbf{A})$, we proceed as in Section 4.4.3 and get

$$\|S_0\|_2 \leq \left\| \int_{\Gamma_0} \mu(\zeta(s), \mathbf{A})\, ds \right\|_2 \leq \left\| \int_{\partial W(\mathbf{A})} \mu(\zeta(s), \mathbf{A})\, ds \right\|_2 = \|2\mathbf{I}\|_2 = 2.$$

Similarly, since $\mu(\zeta(s), \mathbf{A}) + \gamma(s)\mathbf{I}$ is PSD on Γ_1 (thanks to the definition of $\gamma(s)$) and $\mu(\zeta(s), \mathbf{A})$ is PSD on $\partial W(\mathbf{A})$, if we let Γ_2 denote the discarded portion of $\partial W(\mathbf{A})$ (such that $\Gamma_0 \cup \Gamma_2 = \partial W(\mathbf{A})$) and set $\gamma(s)$ to zero on Γ_2, then we get

$$\begin{aligned} \|S_1\|_2 &\leq \left\| \int_{\Gamma_1} (\mu(\zeta(s), \mathbf{A}) + \gamma(s)\mathbf{I})\, ds \right\|_2 \\ &\leq \left\| \int_{\Gamma_1 \cup \Gamma_2} (\mu(\zeta(s), \mathbf{A}) + \gamma(s)\mathbf{I})\, ds \right\|_2 \\ &= \left| \int_{\Gamma_1 \cup \Gamma_2} \gamma(s) \right| = \int_{\Gamma_1} |\gamma(s)|\, ds. \end{aligned}$$

Finally, we get

$$\|S_2\|_2 \leq \int_{\Gamma_1} |\gamma(s)|\, ds.$$

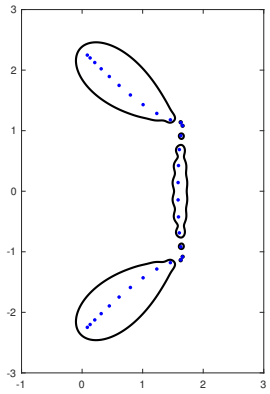


Figure 5.2: For the matrix from the MATLAB command `gallery('grcar',32)`, the eigenvalues (dots) and the boundary components of the 10^{-3}-pseudospectrum (solid curves) are plotted. Direct application of Theorem 4.3 shows that this is a 4.20×10^3-spectral set for $\mathbf{A}$, but (4.8) shows this is only a 2.12×10^3-spectral set. These two values of K are within a factor of approximately 4 from each other.

Since $S(f, \mathbf{A}) = S_0 + S_1 + S_2$, it follows that $\|S(f, \mathbf{A})\|_2 \leq 2 + 2\int_{\Gamma_1} |\gamma(s)|\, ds$ and therefore

$$c_2 \leq 1 + \int_{\Gamma_1} |\gamma(s)|\, ds. \tag{5.2}$$

We can also obtain bounds on K for general Ω that are not obtained by modifying the numerical range. Suppose a set Ω consists of m disjoint, simply connected regions $\Omega_1, \ldots, \Omega_m$ with boundaries $\Gamma_1, \ldots, \Gamma_m$. One example could be the union of disks about each diagonal element of a square matrix defined by the Gershgorin circle theorem. Even if $\mathbf{A}$ has n distinct eigenvalues, if any of these disks about one or more eigenvalues overlap, then $m < n$ and the simply connected regions may not be convex. Another example of Ω with multiple disjoint regions is the 2-norm ϵ-*pseudospectrum* of $\mathbf{A}$:

$$\Lambda_\epsilon(\mathbf{A}) := \{z \in \mathbb{C} : \|(z\mathbf{I} - \mathbf{A})^{-1}\|_2 > \epsilon^{-1}\}.$$

For the set defined by the closure of the ϵ-pseudospectrum, the K value from (4.8) is easy

to compute:

$$K = \frac{\mathcal{L}(\partial\Lambda_\epsilon)}{2\pi\epsilon},$$

where $\mathcal{L}$ maps to the arc length of a curve. To compute a bound on K using Theorem 4.3, we need to bound c_1 and c_2. To bound c_1, it may be difficult to come up with an analytic expression for (4.15), especially when Ω is not a convex set with up to a single disk removed. For example, it is not clear how one could tightly bound c_1 without numerical computations for the ϵ-pseudospectral set in Figure 5.2. However, a bound on c_1 can be estimated numerically (to any desired accuracy). First, we discretize $\partial\Lambda_\epsilon(\mathbf{A})$. Then, we consider each discretization point as a possible value for ζ_0 in (4.15). Next, we determine the total variation of the argument of $\zeta(s) - \zeta_0$ as $\zeta(s)$ traverses the discretized $\partial\Lambda_\epsilon(\mathbf{A})$. Finally, we take c_1 to be $\frac{1}{\pi}$ times the maximum total variation for various ζ_0. To bound c_2, let f be any rational function where $\|f\|_{\Lambda_\epsilon(\mathbf{A})} \leq 1$, and write $S(f, \mathbf{A}) = S_1 + S_2$, where

$$S_1 = \int_{\cup_j \Gamma_j} f(\zeta(s))(\mu(\zeta(s), \mathbf{A}) + \gamma(s)\mathbf{I})\, ds, \quad S_2 = -\int_{\cup_j \Gamma_j} f(\zeta(s))\gamma(s)\mathbf{I}\, ds.$$

Taking $\gamma(s)$ to be greater than or equal to $-\lambda_{min}(\mu(\zeta(s), \mathbf{A}))$ so that $\mu(\zeta(s), \mathbf{A}) + \gamma(s)\mathbf{I}$ is PSD, we get

$$\|S_1\|_2 \leq \left\| \int_{\cup_j \Gamma_j} (\mu(\zeta(s), \mathbf{A}) + \gamma(s)\mathbf{I})\, ds \right\|_2 \leq 2 + \left\| \int_{\cup_j \Gamma_j} \gamma(s)\mathbf{I}\, ds \right\|_2 \leq 2 + \int_{\cup_j \Gamma_j} |\gamma(s)|\, ds,$$

and similarly,

$$\|S_2\|_2 \leq \int_{\cup_j \Gamma_j} |\gamma(s)|\, ds.$$

In this case, $\|S(f, \mathbf{A})\|_2 \leq 2 + 2\int_{\cup_j \Gamma_j} |\gamma(s)|\, ds$ and therefore

$$c_2 \leq 1 + \int_{\cup_j \Gamma_j} |\gamma(s)|\, ds.$$

After the values of c_1 and c_2 are numerically estimated, Theorem 4.3 is used to compute K. While the example used in this section was an ϵ-pseudospectrum set, the same

methodology can be used to numerically estimate the upper bound on K for any subset of the complex plane containing the eigenvalues of $\mathbf{A}$.

5.4 Relationship Between K Values from Theorem 4.3 and Equation (4.8)

Recall the definition of $\mu(\zeta(s), \mathbf{A})$ in (4.13), which we also write as $\mu(\zeta_0, \zeta_0', \mathbf{A})$ for $\zeta_0 = \zeta(s_0)$ and $\zeta_0' = \frac{d\zeta}{ds}\Big|_{s_0}$. Since the magnitude of ζ_0' is 1, it can be written as $e^{i\theta_0}$ for $\theta_0 \in [0, 2\pi)$. Using definition (4.13), we write

$$\mu(\zeta_0, \zeta_0', \mathbf{A}) = \frac{1}{2\pi}\left[e^{i(\theta_0-\pi/2)}(\zeta_0\mathbf{I}-\mathbf{A})^{-1} + e^{-i(\theta_0-\pi/2)}\left((\zeta_0\mathbf{I}-\mathbf{A})^{-1}\right)^*\right]. \tag{5.3}$$

From equation (5.3), we can see that $\mu(\zeta_0, \zeta_0', \mathbf{A})$ is equivalent to $1/\pi$ times the Hermitian part of the matrix $e^{i(\theta_0-\pi/2)}(\zeta_0\mathbf{I}-\mathbf{A})^{-1}$. For any matrix, the minimum eigenvalue of the Hermitian part of the matrix, is equal to the minimum real part of the numerical range of the matrix. From this relation, it is clear that $\lambda_{min}(\mu(\zeta_0, \zeta_0', \mathbf{A}))$ is $\frac{1}{\pi}$ times the smallest real part of points in $W(e^{i(\theta_0-\pi/2)}(\zeta_0\mathbf{I}-\mathbf{A})^{-1})$. We conclude that $|\lambda_{min}(\mu(\zeta_0, \zeta_0', \mathbf{A}))|$ is less than or equal to $\frac{1}{\pi}$ times the numerical radius of $e^{i(\theta_0-\pi/2)}(\zeta_0\mathbf{I}-\mathbf{A})^{-1}$, which is the same as $\frac{1}{\pi}$ times the numerical radius of the resolvent $(\zeta_0\mathbf{I}-\mathbf{A})^{-1}$.

Recall that $\gamma(s) \geq \max\{-\lambda_{min}(\mu(\zeta(s), \mathbf{A})),\ 0\}$. Then in some cases, $\gamma(\zeta_0)$ may be *much* less than $\frac{1}{\pi}$ times the numerical radius of the resolvent. For example, when ζ_0 lies on $\partial W(\mathbf{A})$ so that $\gamma(\zeta_0) = 0$. When $\gamma(\zeta_0)$ is *much* less than $\frac{1}{\pi}$ times the numerical radius of the resolvent, one can expect a *much* smaller value of K in Theorem 4.3 than in (4.8). We can expect the smaller K value because the quantity c_1 is usually of modest size and $2c_2$ will be much less than the value in (4.8). If c_2 is significantly larger than c_1, then the expression for K in Theorem 4.3 is approximately equal to $2c_2$:

$$K = c_2 + c_2\sqrt{1+\frac{c_1}{c_2^2}} = 2c_2 + \frac{1}{2}\frac{c_1}{c_2} + c_2\ O\left(\left(\frac{c_1}{c_2^2}\right)^2\right). \tag{5.4}$$

In other cases, the K value in Theorem 4.3 could exceed that in (4.8) by a factor of

$4 + O(c_1/c_2)$, but no more. To see this, recall from Section 5.3 and equation (5.2) that c_2 is bounded by one plus the integral of $|\lambda_{min}(\mu(\zeta_0, \zeta_0', \mathbf{A}))|$. As previously established, the integrand in equation (5.2) is bounded by $\frac{1}{\pi}$ times the numerical radius of the resolvent $(\zeta_0\mathbf{I} - \mathbf{A})^{-1}$. Since the numerical radius is between $\frac{1}{2}$ and 1 times the norm of the resolvent, c_2 can be bounded by one plus $\frac{1}{\pi}$ times the integral of the resolvent, which is a factor of 2 greater than the formula in (4.8). If the bound on K in Theorem 4.3 is approximately $2c_2$, then this implies the K value from Theorem 4.3 is approximately 4 times greater than the K value from equation (4.8). At most, the K value in Theorem 4.3 could exceed the K value in (4.8) by a factor of $4 + O(c_1/c_2)$, which can be seen from equation (5.4).

We will see in Section 5.5 that in many problems of interest we find that $|\lambda_{min}(\zeta_0, \zeta_0', \mathbf{A})| \approx \frac{1}{\pi} w((\zeta_0\mathbf{I} - \mathbf{A})^{-1})$, and the bound on K in (4.8) is somewhat smaller than that in Theorem 4.3. These problems of interest have a matrix $\mathbf{A}$ that is highly non-normal and a point ζ_0 on the boundary of Ω that comes close to some ill-conditioned eigenvalue(s) of $\mathbf{A}$. We do not yet have a complete explanation of this phenomenon, but here we give an indication of why this might be expected.

5.4.1 When the Numerical Range of the Resolvent is Close to A Disk about a Point Near the Origin

First, note that if $\mathbf{x}$ and $\mathbf{y}$ are two orthogonal unit vectors, then the numerical range of the rank one matrix $\mathbf{x}\mathbf{y}^*$ is a disk about the origin with radius $\frac{1}{2}$. To see this, consider a unitary similarity transformation $\mathbf{Q}^*\mathbf{x}\mathbf{y}^*\mathbf{Q}$, where the columns of $\mathbf{Q}$ are $[\mathbf{x}, \mathbf{y}, \mathbf{q}_3, \ldots, \mathbf{q}_n]$. The matrix $\mathbf{Q}^*\mathbf{x}\mathbf{y}^*\mathbf{Q}$ is the direct sum of a 2 by 2 Jordan block with eigenvalue 0 and an $n-2$ by $n-2$ block of zeros. The numerical range of this matrix is a disk about the origin of radius $\frac{1}{2}$. Note also that the 2-norm of this matrix is 1, which is twice the numerical radius. When $\mathbf{x}$ and $\mathbf{y}$ are *almost* orthogonal to each other, then we use the following theorem that modifies this argument.

Theorem 5.2:

Let $\mathbf{x}$ and $\mathbf{y}$ be unit vectors. Then the rank one matrix $\mathbf{x}\mathbf{y}^*$ is unitarily similar to the direct sum of the 2 by 2 matrix

$$\begin{bmatrix} \frac{1}{2}(\mathbf{y}^*\mathbf{x}) & 1 \\ 0 & \frac{1}{2}(\mathbf{y}^*\mathbf{x}) \end{bmatrix} + \mathbf{E}, \tag{5.5}$$

and an $n-2$ by $n-2$ block of zeros. In equation (5.5), the entries of $\mathbf{E}$ have magnitude $O(|\mathbf{y}^*\mathbf{x}|^2)$. The numerical range of the first matrix in (5.5) is a disk of radius $\frac{1}{2}$ about $\frac{1}{2}(\mathbf{y}^*\mathbf{x})$, and its norm is $1 + O(|\mathbf{y}^*\mathbf{x}|^2)$.

Proof. Let

$$\begin{aligned} \mathbf{q}_1 &= \left(\mathbf{x} - \frac{1}{2}(\mathbf{y}^*\mathbf{x})\mathbf{y}\right) / \left\|\mathbf{x} - \frac{1}{2}(\mathbf{y}^*\mathbf{x})\mathbf{y}\right\|_2, \\ \tilde{\mathbf{q}}_2 &= \left(\mathbf{y} - \frac{1}{2}(\mathbf{x}^*\mathbf{y})\mathbf{x}\right) / \left\|\mathbf{y} - \frac{1}{2}(\mathbf{x}^*\mathbf{y})\mathbf{x}\right\|_2, \\ \mathbf{q}_2 &= (\tilde{\mathbf{q}}_2 - (\mathbf{q}_1^*\tilde{\mathbf{q}}_2)\mathbf{q}_1)/\|\tilde{\mathbf{q}}_2 - (\mathbf{q}_1^*\tilde{\mathbf{q}}_2)\mathbf{q}_1\|_2, \end{aligned}$$

and let $\mathbf{q}_3, \ldots, \mathbf{q}_n$ be any orthonormal vectors that are orthogonal to $\mathbf{q}_1$ and $\mathbf{q}_2$ (and hence to $\mathbf{x}$ and $\mathbf{y}$). Note that

$$\tilde{\mathbf{q}}_2^*\mathbf{q}_1 = \frac{\frac{1}{4}(\mathbf{y}^*\mathbf{x})|\mathbf{y}^*\mathbf{x}|^2}{1 - \frac{3}{4}|\mathbf{y}^*\mathbf{x}|^2},$$

so that $\mathbf{q}_2^*\mathbf{x}$ and $\mathbf{y}^*\mathbf{q}_2$ differ from $\tilde{\mathbf{q}}_2^*\mathbf{x}$ and $\mathbf{y}^*\tilde{\mathbf{q}}_2$ by at most terms of order $|\mathbf{y}^*\mathbf{x}|^3$. Let $\mathbf{Q}$ be the unitary matrix with columns $[\mathbf{q}_1, \ldots, \mathbf{q}_n]$. Then $\mathbf{Q}^*\mathbf{x}\mathbf{y}^*\mathbf{Q}$ is the direct sum of a 2 by 2 matrix and an $n-2$ by $n-2$ block of zeros, where the 2 by 2 matrix is

$$\begin{aligned} \begin{bmatrix} \mathbf{q}_1^* \\ \mathbf{q}_2^* \end{bmatrix} \mathbf{x}\mathbf{y}^*[\mathbf{q}_1, \mathbf{q}_2] &= \begin{bmatrix} (\mathbf{q}_1^*\mathbf{x})(\mathbf{y}^*\mathbf{q}_1) & (\mathbf{q}_1^*\mathbf{x})(\mathbf{y}^*\mathbf{q}_2) \\ (\mathbf{q}_2^*\mathbf{x})(\mathbf{y}^*\mathbf{q}_1) & (\mathbf{q}_2^*\mathbf{x})(\mathbf{y}^*\mathbf{q}_2) \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{2}(\mathbf{y}^*\mathbf{x}) & 1 \\ 0 & \frac{1}{2}(\mathbf{y}^*\mathbf{x}) \end{bmatrix} + \mathbf{E}, \end{aligned}$$

where a straightforward calculation shows that each entry of $\mathbf{E}$ is of order $|\mathbf{y}^*\mathbf{x}|^2$. □

Theorem 5.2 shows that the numerical range of the rank one matrix $\mathbf{x}\mathbf{y}^*$ is close to a disk when $|\mathbf{y}^*\mathbf{x}| << 1$. This disk is centered about the point $\frac{1}{2}(\mathbf{y}^*\mathbf{x})$ whose absolute value is much less than the radius of the disk.

Now, let $\mathbf{A}$ be a diagonalizable matrix with eigenvalues $\lambda_1, \ldots, \lambda_n$ and normalized right and left eigenvectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$ and $\mathbf{y}_1, \ldots, \mathbf{y}_n$. Then the resolvent $(\zeta\mathbf{I} - \mathbf{A})^{-1}$ can be written in the form:

$$(\zeta\mathbf{I} - \mathbf{A})^{-1} = \sum_{j=1}^{n} \frac{1}{\zeta - \lambda_j} \frac{\mathbf{x}_j \mathbf{y}_j^*}{\mathbf{y}_j^* \mathbf{x}_j}. \tag{5.6}$$

If ζ is *much* closer to one eigenvalue, say λ_1, than any other eigenvalues, then the first term in equation (5.6) above will be the largest and

$$(\zeta\mathbf{I} - \mathbf{A})^{-1} \approx \frac{1}{\zeta - \lambda_1} \frac{\mathbf{x}_1 \mathbf{y}_1^*}{\mathbf{y}_1^* \mathbf{x}_1}. \tag{5.7}$$

If λ_1 is ill-conditioned so that $|\mathbf{y}_1^*\mathbf{x}_1| << 1$, then from Theorem 5.2 the numerical range of $(\zeta\mathbf{I} - \mathbf{A})^{-1}$ is approximately equal to $1/((\zeta - \lambda)(\mathbf{y}_1^*\mathbf{x}_1))$ times a disk of radius $\frac{1}{2}$ about the point $\frac{1}{2}(\mathbf{y}_1^*\mathbf{x}_1)$. Thus each point on the boundary of the numerical range of the resolvent has absolute value approximately equal to the numerical radius of the resolvent at points near a simple ill-conditioned eigenvalue.

However, when ζ is close to several ill-conditioned eigenvalues, the approximate equality (5.7) may not hold because other nearby eigenvalues have a tangible contribution to the sum in equation (5.6). The closest rank one matrix to the resolvent $(\zeta\mathbf{I} - \mathbf{A})^{-1}$ in the 2 and Frobenius norms is $\sigma_1 \mathbf{u}_1 \mathbf{v}_1^*$. The rank one matrix is defined using $\sigma_1\left((\zeta\mathbf{I} - \mathbf{A})^{-1}\right)$, the largest *singular value* of $(\zeta\mathbf{I} - \mathbf{A})^{-1}$, and $\mathbf{u}_1$ and $\mathbf{v}_1$ which are the associated left and right singular vectors, respectively. If $\mathbf{u}_1$ and $\mathbf{v}_1$ are almost orthogonal to each other, then using the same reasoning as before $|\gamma(\zeta)| = |\lambda_{\min}(\mu(\zeta, \mathbf{A}))| \approx w\left((\zeta\mathbf{I} - \mathbf{A})^{-1}\right)$, as long as the resolvent can be reasonably approximated with a rank one matrix.

To show that the right and left singular vectors corresponding to the largest singular value of $(\zeta\mathbf{I} - \mathbf{A})^{-1}$ are almost orthogonal to each other when ζ is close to a simple but

ill-conditioned eigenvalue of $\mathbf{A}$, λ, we can use a theorem of G. W. Stewart [101]. First, note that the matrix $\lambda\mathbf{I}-\mathbf{A}$ has a null space of dimension one, and second note that the left and right singular vectors corresponding to the largest singular value of $(\zeta\mathbf{I}-\mathbf{A})^{-1}$ are equivalent to the left and right singular vectors corresponding to the *smallest* singular value of $\zeta\mathbf{I}-\mathbf{A}$. The normalized right and left eigenvectors, $\mathbf{x}$ and $\mathbf{y}$, corresponding to λ satisfy $(\lambda\mathbf{I}-\mathbf{A})\mathbf{x}=0$ and $(\lambda\mathbf{I}-\mathbf{A})^*\mathbf{y}=0$. It follows that these are right and left singular vectors of $\lambda\mathbf{I}-\mathbf{A}$ corresponding to the smallest singular value, $\sigma_n(\lambda\mathbf{I}-\mathbf{A})=0$. We write the SVD of $\lambda\mathbf{I}-\mathbf{A}$ as $\mathbf{Y}\Sigma\mathbf{X}^*$, where $\mathbf{X}=[\mathbf{x},\mathbf{X}_2]$ and $\mathbf{Y}=[\mathbf{y},\mathbf{Y}_2]$, and the singular values ordered in increasing order starting with σ_n. Define $\mathbf{E}:=(\zeta-\lambda)\mathbf{I}$ so that $(\lambda\mathbf{I}-\mathbf{A})+\mathbf{E}=\zeta\mathbf{I}-\mathbf{A}$. Define

$$\iota := \left\| \begin{bmatrix} \mathbf{Y}_2^*\mathbf{E}\mathbf{x} \\ \mathbf{X}_2^*\mathbf{E}^*\mathbf{y} \end{bmatrix} \right\|_F = \left\| \begin{bmatrix} (\zeta-\lambda)\mathbf{Y}_2^*\mathbf{x} \\ (\bar{\zeta}-\bar{\lambda})\mathbf{X}_2^*\mathbf{y} \end{bmatrix} \right\|_F \leq \sqrt{2}\,|\zeta-\lambda|,$$

$$\begin{aligned}\delta :=& \sigma_{n-1}(\lambda\mathbf{I}-\mathbf{A}) - \|\mathbf{y}^*\mathbf{E}\mathbf{x}\|_2 - \|\mathbf{Y}_2^*\mathbf{E}\mathbf{X}_2\|_2 \\ =& \sigma_{n-1}(\lambda\mathbf{I}-\mathbf{A}) - |\zeta-\lambda|\left(|\mathbf{y}^*\mathbf{x}| + \|\mathbf{Y}_2^*\mathbf{X}_2\|_2\right) \\ \geq& \sigma_{n-1}(\lambda\mathbf{I}-\mathbf{A}) - |\zeta-\lambda|(1+|\mathbf{y}^*\mathbf{x}|),\end{aligned}$$

where $\sigma_{n-1}(\lambda\mathbf{I}-\mathbf{A})$ is the second smallest singular value of $\lambda\mathbf{I}-\mathbf{A}$. Assuming that $\iota/\delta<1/2$, it is shown in [101, Theorem 6.4] that there are vectors $\mathbf{p}$ and $\mathbf{q}$ satisfying

$$\left\| \begin{bmatrix} \mathbf{p} \\ \mathbf{q} \end{bmatrix} \right\|_F < 2\frac{\iota}{\delta}$$

such that $(\mathbf{x}+\mathbf{X}_2\mathbf{p})/\|\mathbf{x}+\mathbf{X}_2\mathbf{p}\|_2$ and $(\mathbf{y}+\mathbf{Y}_2\mathbf{q})/\|\mathbf{y}+\mathbf{Y}_2\mathbf{q}\|_2$ are the right and left singular vectors of $(\lambda\mathbf{I}-\mathbf{A})+\mathbf{E}=\zeta\mathbf{I}-\mathbf{A}$ corresponding to the smallest singular value. Then $\mathbf{x}+\mathbf{X}_2\mathbf{p}$ and $\mathbf{y}+\mathbf{Y}_2\mathbf{q}$ are equivalently multiples of the left and right singular vectors of $(\zeta\mathbf{I}-\mathbf{A})^{-1}$ corresponding to the largest singular value. It follows that if $\mathbf{x}$ and $\mathbf{y}$ are almost orthogonal to each other and if $\|\mathbf{p}\|_2$ and $\|\mathbf{q}\|_2$ are small, then these singular vectors (corresponding to

the largest singular value of $(\zeta\mathbf{I}-\mathbf{A})^{-1}$) are almost orthogonal to each other:

$$\left|\frac{(\mathbf{x}+\mathbf{X}_2\mathbf{p})^*(\mathbf{y}+\mathbf{Y}_2\mathbf{q})}{\|\mathbf{x}+\mathbf{X}_2\mathbf{p}\|_2\|\mathbf{y}+\mathbf{Y}_2\mathbf{q}\|_2}\right| = \frac{|\mathbf{x}^*\mathbf{y}+\mathbf{x}^*\mathbf{Y}_2\mathbf{q}+\mathbf{p}^*\mathbf{X}_2^*\mathbf{y}+\mathbf{p}^*\mathbf{X}_2^*\mathbf{Y}_2\mathbf{q}|}{\|\mathbf{x}+\mathbf{X}_2\mathbf{p}\|_2\|\mathbf{y}+\mathbf{Y}_2\mathbf{q}\|_2} \leq \frac{|\mathbf{x}^*\mathbf{y}|+\|\mathbf{q}\|_2+\|\mathbf{p}\|_2+\|\mathbf{p}\|_2\|\mathbf{q}\|_2}{\sqrt{(1-\|\mathbf{p}\|_2^2)(1-\|\mathbf{q}\|_2^2)}}.$$

Now that we have shown that the rank one matrix, $\sigma_1\mathbf{u}_1\mathbf{v}_1^*$, is defined by nearly orthogonal vectors, we need to show when the resolvent is reasonably approximated by that matrix. The 2-norm of $(\zeta\mathbf{I}-\mathbf{A})^{-1}-\sigma_1\mathbf{u}_1\mathbf{v}_1^*$ is equal to $\sigma_2\left((\zeta\mathbf{I}-\mathbf{A})^{-1}\right)$, the second largest singular value. So $(\zeta\mathbf{I}-\mathbf{A})^{-1}\approx\sigma_1\mathbf{u}_1\mathbf{v}_1^*$ when $\sigma_2\left((\zeta\mathbf{I}-\mathbf{A})^{-1}\right)$ is relatively small. We bound the relative difference between $\sigma_2((\zeta\mathbf{I}-\mathbf{A})^{-1})$ and $\sigma_1((\zeta\mathbf{I}-\mathbf{A})^{-1})$ by perturbing the singular values of $\lambda\mathbf{I}-\mathbf{A}$, where λ is an eigenvalue of $\mathbf{A}$ nearby ζ. One may expect the singular values of $\zeta\mathbf{I}-\mathbf{A}$ to differ from those of $\lambda\mathbf{I}-\mathbf{A}$ by $O(|\zeta-\lambda|)$ [74, Theorem 3.3.16]. However, in the special case where λ is an ill-conditioned eigenvalue of $\mathbf{A}$ and $|\zeta-\lambda|$ is much less than the second smallest singular value of $\lambda\mathbf{I}-\mathbf{A}$, $\sigma_{n-1}(\lambda\mathbf{I}-\mathbf{A})$, we show that the change in the smallest singular value, $\sigma_n(\lambda\mathbf{I}-\mathbf{A})=0$, is actually much less than $|\zeta-\lambda|$.

Theorem 5.3:

Let λ be a simple eigenvalue of $\mathbf{A}$ and let $\lambda\mathbf{I}-\mathbf{A}=\mathbf{U}\mathbf{\Sigma}\mathbf{V}^*$ be a singular value decomposition of $\lambda\mathbf{I}-\mathbf{A}$, with $\mathbf{U}:=[\mathbf{u}_1,\cdots,\mathbf{u}_n]$, $\mathbf{V}:=[\mathbf{v}_1,\cdots,\mathbf{v}_n]$, and $\mathbf{\Sigma}:=\mathrm{diag}(\sigma_1,\cdots,\sigma_{n-1},0)$. Then the smallest singular value of $\zeta\mathbf{I}-\mathbf{A}$ is less than or equal to

$$|\zeta-\lambda|\ |\mathbf{u}_n^*\mathbf{v}_n|+\frac{|\zeta-\lambda|^2}{\sigma_{n-1}}. \tag{5.8}$$

Proof. Let $\mathbf{U}_{n-1}:=[\mathbf{u}_1,\cdots,\mathbf{u}_{n-1}]$, $\mathbf{V}_{n-1}:=[\mathbf{v}_1,\cdots,\mathbf{v}_{n-1}]$, and $\mathbf{\Sigma}_{n-1}:=\mathrm{diag}(\sigma_1,\cdots,\sigma_{n-1})$.

Then

$$\begin{aligned}(\zeta\mathbf{I}-\mathbf{A})&(\mathbf{v}_n-(\zeta-\lambda)\mathbf{V}_{n-1}\mathbf{\Sigma}_{n-1}^{-1}\mathbf{U}_{n-1}^*\mathbf{v}_n)\\ &=(\lambda\mathbf{I}-\mathbf{A}+(\zeta-\lambda)\mathbf{I})(\mathbf{v}_n-(\zeta-\lambda)\mathbf{V}_{n-1}\mathbf{\Sigma}_{n-1}^{-1}\mathbf{U}_{n-1}^*\mathbf{v}_n)\\ &=(\lambda\mathbf{I}-\mathbf{A})\mathbf{v}_n-(\zeta-\lambda)\big((\lambda\mathbf{I}-\mathbf{A})\mathbf{V}_{n-1}\big)\mathbf{\Sigma}_{n-1}^{-1}\mathbf{U}_{n-1}^*\mathbf{v}_n\\ &\quad+(\zeta-\lambda)(\mathbf{v}_n-(\zeta-\lambda)\mathbf{V}_{n-1}\mathbf{\Sigma}_{n-1}^{-1}\mathbf{U}_{n-1}^*\mathbf{v}_n)\\ &=(\zeta-\lambda)(\mathbf{v}_n-\mathbf{U}_{n-1}\mathbf{U}_{n-1}^*\mathbf{v}_n-(\zeta-\lambda)\mathbf{V}_{n-1}\mathbf{\Sigma}_{n-1}^{-1}\mathbf{U}_{n-1}^*\mathbf{v}_n).\end{aligned}$$

Since the vector $\mathbf{v}_n-\mathbf{U}_{n-1}\mathbf{U}_{n-1}^*\mathbf{v}_n$ is the orthogonal projection onto the span of $\mathbf{u}_n$, it can be rewritten as $\mathbf{u}_n\mathbf{u}_n^*\mathbf{v}_n$. The vector $\mathbf{u}_n\mathbf{u}_n^*\mathbf{v}_n$ has norm equal to $|\mathbf{u}_n^*\mathbf{v}_n|$. Further, $\|\mathbf{V}_{n-1}\mathbf{\Sigma}_{n-1}^{-1}\mathbf{U}_{n-1}^*\mathbf{v}_n\|_2\leq\sigma_{n-1}^{-1}$. Combining these facts, it follows that

$$\|(\zeta\mathbf{I}-\mathbf{A})(\mathbf{v}_n-(\zeta-\lambda)\mathbf{V}_{n-1}\mathbf{\Sigma}_{n-1}^{-1}\mathbf{U}_{n-1}^*\mathbf{v}_n)\|_2\leq|\zeta-\lambda|\ |\mathbf{u}_n\mathbf{v}_n^*|+\frac{|\zeta-\lambda|^2}{\sigma_{n-1}},\tag{5.9}$$

and since the vector $\mathbf{v}_n-(\zeta-\lambda)\mathbf{V}_{n-1}\mathbf{\Sigma}_{n-1}^{-1}\mathbf{U}_{n-1}^*\mathbf{v}_n$ has norm greater than or equal to the norm of $\mathbf{v}_n$, which is 1, the result in (5.8) follows. □

Meanwhile, the second smallest singular value, $\sigma_{n-1}(\lambda\mathbf{I}-\mathbf{A})$, decreases by at most $|\zeta-\lambda|$, and may actually increase by that amount. The fact that $\sigma_n(\lambda\mathbf{I}-\mathbf{A})$ increases according to equation (5.8), which is much less than $|\zeta-\lambda|$, and that $\sigma_{n-1}(\lambda\mathbf{I}-\mathbf{A})$ decreases by at most $|\zeta-\lambda|$ when ζ is near an ill-conditioned eigenvalue, causes us to conclude that the relative difference of the resolvent and the rank one matrix defined by the smallest singular value of $\zeta\mathbf{I}-\mathbf{A}$ remains small when $\sigma_{n-1}>>|\zeta-\lambda|$. When the relative difference remains small, we know that $(\zeta\mathbf{I}-\mathbf{A})^{-1}$ is well-approximated by a rank one matrix defined by nearly orthogonal vectors.

Greenbaum, Kyanfar, and Salemi define the 2-norm *relative difference* between the re-

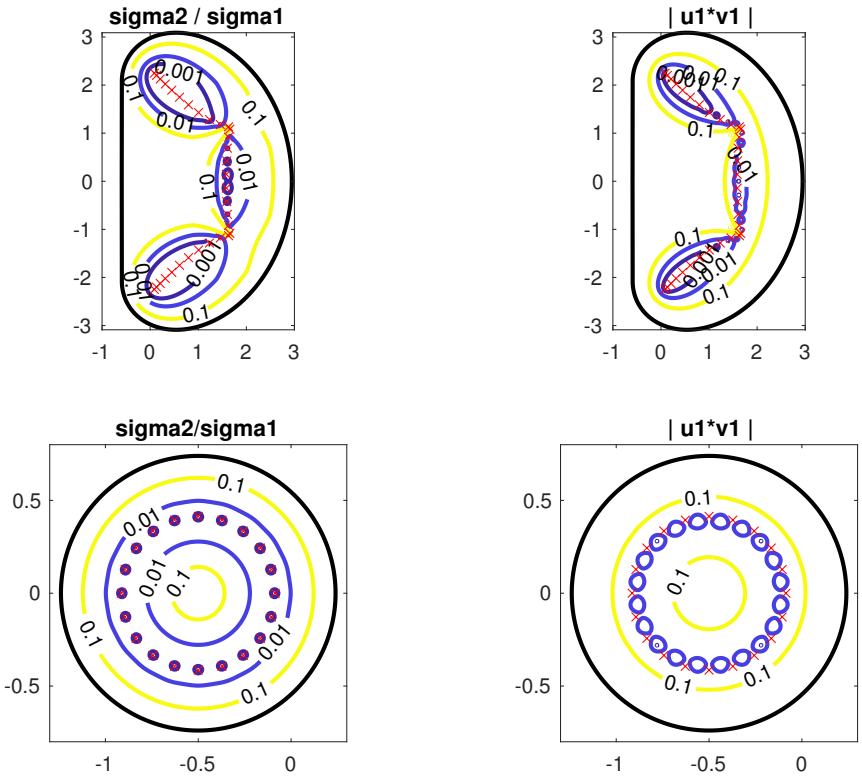


Figure 5.3: The left shows contour plots of $\mathbb{S}_\epsilon(\mathbf{A})$ for different values of ϵ, and the right shows contour plots of the inner products $|u_1^* v_1|$ of left and right singular vectors corresponding to the largest singular value of $(\zeta \mathbf{I} - \mathbf{A})^{-1}$. The top row shows these plots for the Grcar matrix of size $n = 32$ and the bottom shows them for the `transient_demo` matrix of size $n = 20$. Also shown are the eigenvalues (red 'x's) and the boundary of the numerical range (thick black curve).

solvent and this rank one matrix as

$$\frac{\|(\zeta \mathbf{I} - \mathbf{A})^{-1} - \sigma_1\left((\zeta \mathbf{I} - \mathbf{A})^{-1}\right) \mathbf{u}_1 \mathbf{v}_1^*\|_2}{\|(\zeta \mathbf{I} - \mathbf{A})^{-1}\|_2} = \frac{\sigma_2\left((\zeta \mathbf{I} - \mathbf{A})^{-1}\right)}{\sigma_1\left((\zeta \mathbf{I} - \mathbf{A})^{-1}\right)} \tag{5.10}$$

in [58]. To illustrate when the relative difference remains small, we plot the set

$$\mathbb{S}_\epsilon(\mathbf{A}) := \{\zeta \in \mathbb{C} : \sigma_2(\zeta)/\sigma_1(\zeta) < \epsilon\}, \tag{5.11}$$

where $\sigma_1(\zeta)$ and $\sigma_2(\zeta)$ are the largest and second largest singular values of the resolvent

$(\zeta\mathbf{I} - \mathbf{A})^{-1}$ [58]. Figure 5.3 shows contour plots of $\mathbb{S}_\epsilon(\mathbf{A})$ for various values of ϵ on the left, and contour plots of the inner products $|\mathbf{u}_1^*\mathbf{v}_1|$ of the left and right singular vectors corresponding to $\sigma_1(\zeta)$ for two highly non-normal matrices on the right. Note the large areas over which these ratios and inner products are small, implying that the numerical range of the resolvent is close to a disk about a point much nearer to the origin than the radius of the disk.

The top plots in Figure 5.3 are for the Grcar matrix of size $n = 32$. This matrix has -1's on the subdiagonal, 1's on the main diagonal and the first three super-diagonals, and 0's elsewhere. For a Grcar matrix of size $n = 100$, in [29] it was shown that the K value obtained from Theorem 4.3 is much smaller than the K value obtained from (4.8) when the region Ω is taken to be $W(\mathbf{A})\backslash\mathcal{D}(0, 1/w(\mathbf{A}^{-1}))$. Figure 5.3 shows that this will not be the case if one chooses a smaller region Ω; e.g., the 10^{-3} pseudospectrum, pictured in Figure 5.2. The 10^{-3} pseudospectrum looks similar to the 10^{-3} level curve of σ_2/σ_1, so that at points on the boundary of the 10^{-3} pseudospectrum, the resolvent $(\zeta\mathbf{I} - \mathbf{A})^{-1}$ is close to a rank one matrix. The eigenvalues of the Grcar matrix with larger real parts (real parts near about 1.6) are better-conditioned than the other eigenvalues that have condition numbers ranging from about 16 to 137, resulting in smaller regions about these eigenvalues. Condition numbers of the other eigenvalues range from 278 to 3460. The bottom plots in Figure 5.3 are for the `transient_demo` matrix of size 20, available in the eigtool package [120], which will be used again in Section 5.5. The eigenvalues of the `transient_demo` matrix all have the same condition number, about 72.6.

5.5 Applications

Throughout this section we assume the norm of interest is the 2-norm. The MATLAB codes used to produce the results and figures in this section can be found at `https://www.github.com/tygris/k-spectral-sets`.

5.5.1 Block Diagonal Matrices

If $\mathbf{A}$ is a block diagonal matrix, say,

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & 0 \\ 0 & \mathbf{A}_{22} \end{bmatrix},$$

then since

$$f(\mathbf{A}) = \begin{bmatrix} f(\mathbf{A}_{11}) & 0 \\ 0 & f(\mathbf{A}_{22}) \end{bmatrix},$$

it is clear that $\|f(\mathbf{A})\|_2$ can be bounded using the $\max_z |f(z)|$ on $W(\mathbf{A}_{11}) \cup W(\mathbf{A}_{22})$ instead of $\max_z |f(z)|$ on $W(\mathbf{A})$. The numerical range $W(\mathbf{A})$ is likely a larger set since $W(\mathbf{A})$ is equivalent to the convex hull of $W(\mathbf{A}_{11}) \cup W(\mathbf{A}_{22})$. Of course, if we knew *a priori* that $\mathbf{A}$ was block diagonal, we could take advantage of this property and only consider the numerical ranges of the inidividual blocks. When $\mathbf{A}$ is not in block diagonal form but is unitarily similar to a block diagonal matrix the same property holds. To identify the blocks of a matrix unitarily similar to a block diagonal matrix is an NP-hard problem [63]. An alternative to solving this NP-hard problem could be to start with $W(\mathbf{A})$ and remove one or more disks to cut the numerical range into disjoint pieces corresponding to the blocks of $\mathbf{A}$.

An example that removes a disk from the numerical range of a block diagonal matrix is illustrated in Figure 5.4. The first block $\mathbf{A}_{11}$ is a real random 4 by 4 matrix, and the second block $\mathbf{A}_{22}$ is a different real random 4 by 4 matrix plus $8\mathbf{I}$, where the random matrix entries were drawn from a standard normal distribution. The disk removed from the numerical range in the figure was centered at $\xi = 3.5$, and it had radius $1/w((\xi\mathbf{I} - \mathbf{A})^{-1})$. In this case, removing a single disk split the numerical range into two disjoint pieces. According to the results from Crouzeix and Greenbaum (2109) explained in section 4.4.3, the remaining region (outlined with a thick black line in the figure) is a $(3 + 2\sqrt{3})$-spectral set for $\mathbf{A}$ [29], and $(3+2\sqrt{3})$ is approximately 6.46. For comparison, using (4.8) to bound K by integrating

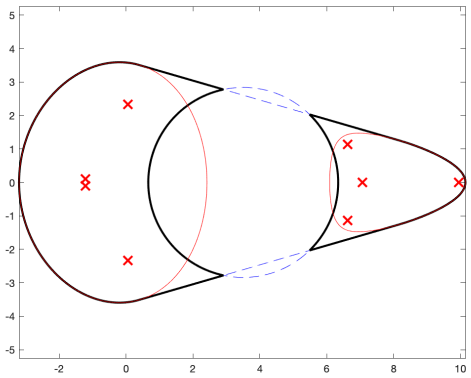


Figure 5.4: The eigenvalues and numerical range of a block diagonal matrix defined by two different 4 by 4 random matrices with entries drawn from a standard normal distribution, with the second block shifted right by $8\mathbf{I}$. The numerical range is cut into two pieces by removing a disk about $\xi = 3.5$ with radius $1/w((\xi\mathbf{I}-\mathbf{A})^{-1})$. The resulting region is outlined in thick black; the numerical ranges of the individual blocks are shown in thin red.

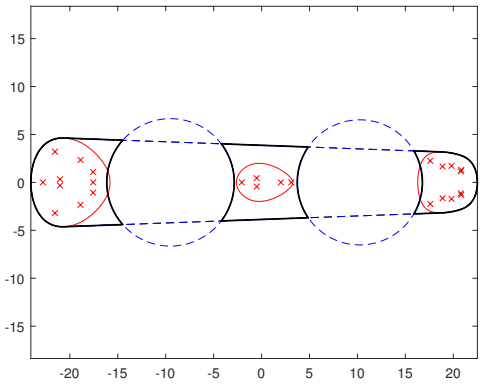


Figure 5.5: $\mathbf{A}$ is a block diagonal matrix with three blocks. Each block is the sum of a multiple of the identity and a real random matrix $\mathbf{R}$ with entries from a standard normal distribution. Block $\mathbf{A}_{11} = -20\mathbf{I} + \mathbf{R}_1$ is 10 by 10, block $\mathbf{A}_{22} = \mathbf{R}_2$ is 5 by 5, and block $\mathbf{A}_{33} = 20\mathbf{I}+\mathbf{R}_3$ is 10 by 10. The disks removed had radii $1/\|(\xi_{1,2}\mathbf{I}-\mathbf{A})^{-1}\|_2$, where $\xi_1 = -9.5$ and $\xi_2 = 10$. The remaining region is a $3 + \sqrt{14} \approx 6.74$-spectral set when calculating K with equation (5.1), and when calculating K with Theorem 4.3 as described in section 5.3, we get $c_1 \leq 2.60$, $c_2 \leq 1.78$, and $K = 4.19$. When calculating K using (4.8), $K = 11.88$.

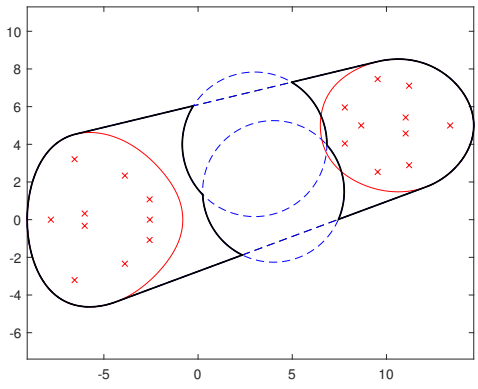


Figure 5.6: $\mathbf{A}$ is a block diagonal matrix with two blocks. Each block is the sum of a multiple of the identity and a real random matrix $\mathbf{R}$ with entries from a standard normal distribution. Block $\mathbf{A}_{11} = -5\mathbf{I} + \mathbf{R}_1$ is 10 by 10, and block $\mathbf{A}_{22} = (10 + 5i)\mathbf{I} + \mathbf{R}_2$ is 10 by 10. Two disks of radius $1/\|(\xi_{1,2}\mathbf{I} - \mathbf{A})^{-1}\|_2$, where $\xi_1 = 4 + 1.5i$ and $\xi_2 = 3 + 4i$, were removed to split the numerical range of $\mathbf{A}$ into two disjoint sets. The remaining region is a $3 + \sqrt{14} \approx 6.74$-spectral set when calculating K with equation (5.1), and when calculating K using Theorem 4.3 as described in section 5.3, $c_1 \leq 3.20$, $c_2 \leq 1.73$, and $K = 4.21$. When calculating K using (4.8), $K = 7.94$.

the resolvent over the boundary of this set, one obtains the slightly larger value of 8.01. Both these bounds on K are the same order of magnitude. One strength of using Theorem 4.3 to bound K is that the bound is known analytically. However, the analytical bound from Theorem 4.3 requires that Ω is defined in a specific way that frequently requires numerical computation. Also shown in Figure 5.4 with a thin red line in the figure are the numerical ranges of each block, and in red 'x's are the eigenvalues of $\mathbf{A}$.

In general, removing one disk from the numerical range may not be enough to split the set into disjoint pieces. For a matrix with more than two diagonal blocks, one could remove two or more disks from $W(\mathbf{A})$ and obtain a K-spectral set with three or more disjoint simply connected regions, where K is bounded by expression (5.1). In other cases, a single disk may not be wide enough to split the numerical range into disjoint pieces. Then multiple disks could be removed, and K would again be bounded by expression (5.1).

Alternatively, a smaller bound on K may be obtained by using Theorem 4.3 directly and

numerically determining the bounds on c_1 and c_2, as described in section 5.3. Figures 5.5 and 5.6 illustrate those two cases respectively and state the K values calculated by formula (5.1) and directly from Theorem 4.3. In all of these cases, the differences between the theoretical value from formula (5.1), the numerical value obtained by directly computing c_1 and c_2 in Theorem 4.3, and the numerical value obtained from equation (4.8) are not large, but directly computing Theorem 4.3 and using formula (5.1) both lead to somewhat smaller K values than (4.8). Thus, the new analysis provides some improvement.

5.5.2 Bounding Solutions to the Initial Value Problem

The results from section 5.3 can be used to bound the solutions to both continuous and discrete time dynamical systems. If the spectrum of $\mathbf{A}$ lies in the left half-plane or the unit disk respectively, then the norm of the dynamical system is bounded by K for the set Ω equal to the intersection of $W(\mathbf{A})$ with the left half-plane or the unit disk, respectively.

As an example, the left plot in Figure 5.7 shows the behavior of $\|e^{t\mathbf{A}}\|_2$ for a matrix $\mathbf{A}$ that models the ecosystem of Tuesday Lake in Wisconsin after the introduction of pisciverous largemouth bass [83]. The plot shows initial growth and then decay of the relative total population of the Tuesday Lake ecosystem as the continuous time parameter t changes. The right plot in the figure shows 'x's for the eigenvalues, a dashed line for the numerical range of the matrix, and a thick black line denoting Ω equal to the numerical range intersected with the left half-plane. By integrating $|\lambda_{min}(\mu(\zeta(s), \mathbf{A})|$ along the segment of the imaginary axis inside $W(\mathbf{A})$ and using Theorem 4.3, we found that $K = 2.66$, while using (4.8) $K = 3.72$. In this case the different bounds on K are all very close and somewhat larger than the maximum value of $\|e^{t\mathbf{A}}\|_2$, $t > 0$ equal to 1.06 found in Figure 5.7.

Another example we consider is the matrix `transient_demo(20)` available in the eigtool package [120]. The upper left plot in Figure 5.8 shows the behavior of $\|e^{t\mathbf{A}}\|_2$, $t > 0$, which grows to about 16.61 before starting to decrease. The upper right plot shows the eigenvalues ('x's) in the left half-plane, and the numerical range (dashed line) extending into

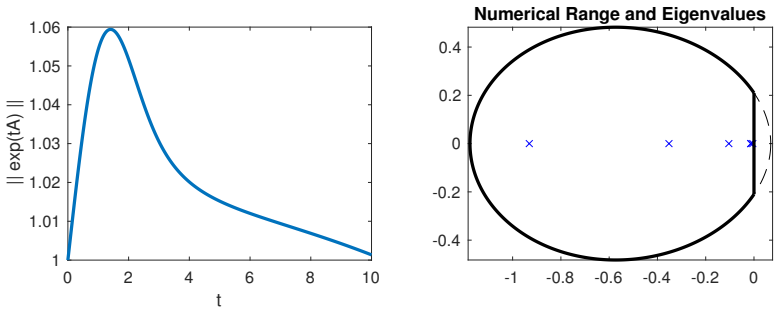


Figure 5.7: The matrix $\mathbf{A}$ is the dynamical system coefficient matrix modeling the ecosystem of Tuesday Lake after introducing piscivores [83]. The left plot shows $\|e^{t\mathbf{A}}\|_2$ growing before decaying; the right plot shows $W(\mathbf{A})$ extending into the right half-plane (dashed curve), eigenvalues ('x's) are in the left half-plane, and K-spectral set Ω (thick black line) with $K = 2.66$ using Theorem 4.3 and $K = 3.72$ using (4.8).

the right half-plane, together with the region Ω (thick black curve) equal to the intersection of $W(\mathbf{A})$ and the left half-plane. By integrating $|\lambda_{min}(\mu(\zeta(s), \mathbf{A}))|$ along the segment of the imaginary axis forming the right boundary of Ω and using Theorem 4.3, we determine that $K = c_2 + \sqrt{c_2^2 + c_1} \approx 2c_2 = 40.13$. Using equation (4.8) the smaller bound $K = 27.95$ is found. The reason for this smaller value can be seen in the lower plots of Figure 5.8. The large values of $|\lambda_{min}(\mu(\zeta(s), \mathbf{A}))|$ and of $\frac{1}{2\pi}\|(\zeta\mathbf{I} - \mathbf{A})^{-1}\|_2$ both occur on the segment of the imaginary axis belonging to $\partial\Omega$ and nearest to an ill-conditioned eigenvalue. As can be seen in the lower left plot, while $|\lambda_{min}(\mu(\zeta(s), \mathbf{A})|$ is always less than or equal to $\frac{1}{2\pi}\|(\zeta\mathbf{I}-\mathbf{A})^{-1}\|_2$, the difference is small. The lower right plot shows the numerical ranges of several of the matrices $\frac{\zeta'}{2\pi i}(\zeta\mathbf{I} - \mathbf{A})^{-1} = \frac{1}{2\pi}(\zeta\mathbf{I} - \mathbf{A})^{-1}$ for ζ equal to $0, \pm 0.2725i$, and $\pm 0.545i$. While the smallest numerical ranges lie mostly in the right half-plane, the absolute value of the real part of the leftmost points in the larger numerical ranges, equal to $\frac{1}{2}|\lambda_{min}(\mu(\zeta, \mathbf{A}))|$, are almost as large as the corresponding numerical radii. The largest numerical range in the lower right plot of Figure 5.8 corresponds to $\zeta = 0$, which is the point on the imaginary axis closest to an ill-conditioned eigenvalue of the `transient_demo(20)` matrix. The value of K from Theorem 4.3 is approximately equal to $2c_2$, which is approximately twice the integral of $|\lambda_{min}(\mu(\zeta(s), \mathbf{A}))|$ over this segment. The value of K from (4.8) is the integral of

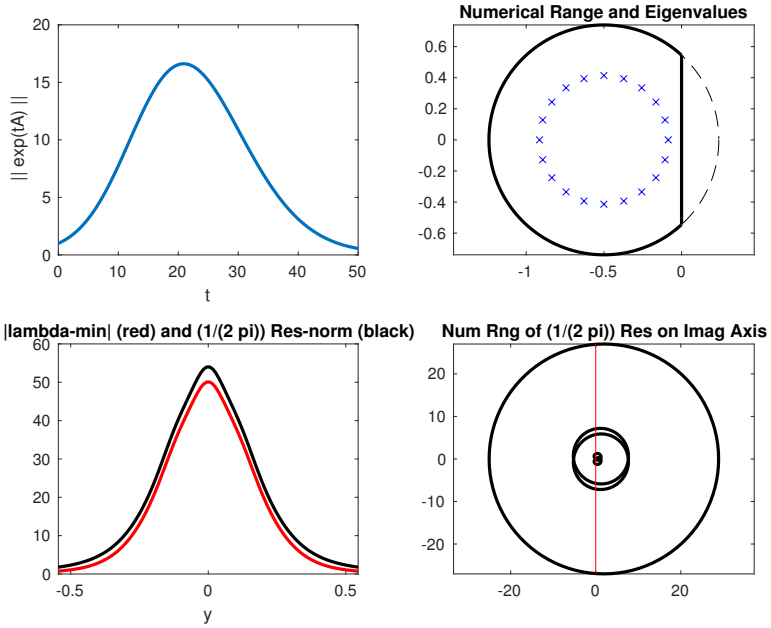


Figure 5.8: The matrix $\mathbf{A}$ comes from the eigtool command `transient_demo(20)` [120]. The upper left plot shows $\|e^{t\mathbf{A}}\|_2$ growing as t increases before decaying. The upper right plot shows $W(\mathbf{A})$ extending into the right half-plane (dashed curve), eigenvalues of $\mathbf{A}$ ('x's) in the left half-plane, and $\partial\Omega$ (thick black) where Ω is the intersection of $W(\mathbf{A})$ and the left half-plane. The lower left plot shows $|\lambda_{\min}(\mu(\zeta), \mathbf{A}))|$ (red) and $\frac{1}{2\pi}\|(\zeta\mathbf{I} - \mathbf{A})^{-1}\|_2$ (black) above the red curve for ζ on the segment of the imaginary axis forming the right-most boundary of Ω. The lower right plot shows numerical ranges of $\frac{1}{2\pi}(\zeta\mathbf{I} - \mathbf{A})^{-1}$ for ζ on the segment of the imaginary axis in $\partial\Omega$. The larger numerical ranges have an absolute value of the minimal real part, which is $\frac{1}{2}|\lambda_{min}(\mu(\zeta, \mathbf{A}))|$, almost as large as the numerical radius. The similarity in size to the numerical radius explains why $|\lambda_{min}(\mu(\zeta, \mathbf{A}))|$ is of the same order of magnitude as $\frac{1}{2\pi}\|(\zeta\mathbf{I} - \mathbf{A})^{-1}\|_2$, as explained in Section 5.4.

$\frac{1}{2\pi}\|(\zeta\mathbf{I} - \mathbf{A})^{-1}\|_2$ over this segment and the remainder of $\partial\Omega$ where $\|(\zeta\mathbf{I} - \mathbf{A})^{-1}\|_2$ is much smaller. Since computing K using Theorem 4.3 is approximately $2c_2$, the result is a smaller value of K using formula (4.8).

In Figures 5.8 and 5.9, a different choice of Ω was considered for the same non-normal matrix $\mathbf{A}$. The choice in Figure 5.9, where Ω is the part of $W(\mathbf{A})$ inside the unit disk, is relevant when studying norms of powers of $\mathbf{A}$. The function $f(z) = z^k$ is bounded in absolute value by 1 for z inside the unit disk, just as $|e^z|$ is bounded by 1 for z in the left half-plane. Using the matrix `transient_demo(20)`, we computed norms of powers of $\mathbf{A}$ and

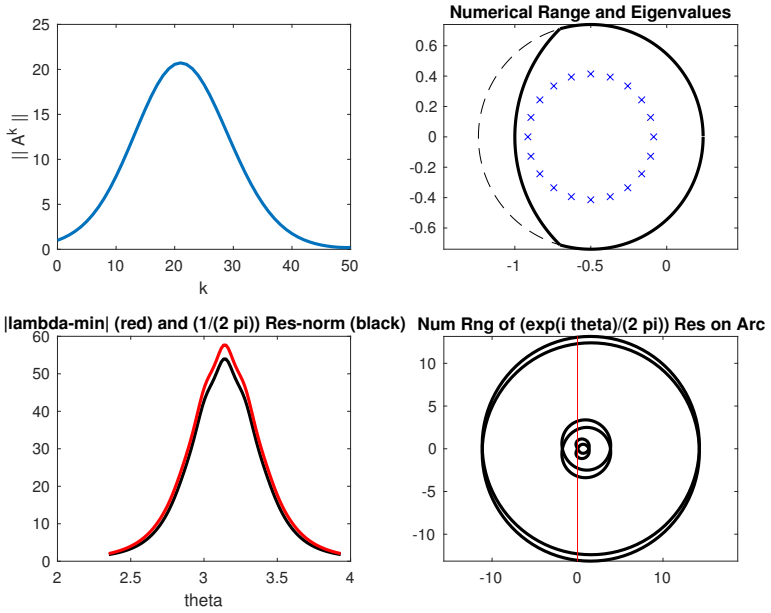


Figure 5.9: The matrix $\mathbf{A}$ comes from the eigtool command `transient_demo(20)` [120]. The upper left plot shows $\|\mathbf{A}^k\|_2$ growing as k increases before decaying. The upper right plot shows $W(\mathbf{A})$ extending beyond $\mathcal{D}(0,1)$ (dashed curve), the eigenvalues of $\mathbf{A}$ ('x's), and $\partial\Omega$ (thick black) where Ω is the intersection of $W(\mathbf{A})$ and the unit disk. The lower left plot shows $|\lambda_{\min}(\mu(\zeta,\mathbf{A}))|$ (red) and $\frac{1}{2\pi}\|(\zeta\mathbf{I}-\mathbf{A})^{-1}\|_2$ (black) below the red curve for ζ on the portion of $\partial\Omega$ belonging to the arc of the unit circle inside $W(\mathbf{A})$. The lower right plot shows numerical ranges of $\frac{e^{i\theta}}{2\pi}(\zeta\mathbf{I}-\mathbf{A})^{-1}$ for $\zeta = e^{i\theta}$ and $\theta = \pm 2.8763, \pm 2.626$, and ± 2.3504. These values of θ are located along the portion of $\partial\Omega$ equivalent to the arc of the unit circle. The larger numerical ranges have an absolute value of the minimal real part, $\frac{1}{2}|\lambda_{min}(\mu(\zeta,\mathbf{A}))|$, that is almost as large as the numerical radius. This similarity explains why $|\lambda_{min}(\mu(\zeta,\mathbf{A}))|$ is of the same order of magnitude as $\frac{1}{2\pi}\|(\zeta\mathbf{I}-\mathbf{A})^{-1}\|_2$.

found that they grew to about 20.72 before starting to decrease, as shown in the upper left plot of Figure 5.9. The upper right plot shows the numerical range of the matrix (dashed), which extends beyond $\mathcal{D}(0,1)$, and the eigenvalues ('x's) which all lie within $\mathcal{D}(0,1)$. The K-spectral set Ω (thick black) is $W(\mathbf{A}) \cap \mathcal{D}(0,1)$. In the lower left plot, we see that there are large values of $|\lambda_{min}(\mu(\zeta,\mathbf{A}))|$ and of $\frac{1}{2\pi}\|(\zeta\mathbf{I}-\mathbf{A})^{-1}\|_2$ on the arc of the unit circle inside $W(\mathbf{A})$. In this example, $|\lambda_{min}(\mu(\zeta(s),\mathbf{A})|$ is greater than $\frac{1}{2\pi}\|(\zeta\mathbf{I}-\mathbf{A})^{-1}\|_2$. The lower right plot shows why $|\lambda_{min}(\mu(\zeta,\mathbf{A}))|$ might be larger than $\frac{1}{2\pi}\|(\zeta\mathbf{I}-\mathbf{A})^{-1}\|_2$. It shows the numerical ranges of several of the matrices $\frac{\zeta'}{2\pi i}(\zeta\mathbf{I}-\mathbf{A})^{-1} = \frac{e^{i\theta}}{2\pi}(\zeta\mathbf{I}-\mathbf{A})^{-1}$ for $\zeta = e^{i\theta}$

when $\theta = \pm 2.8763, \pm 2.626$, and ± 2.3504. The absolute value of the real part of the leftmost points in the larger numerical ranges (which is $\frac{1}{2}|\lambda_{min}(\mu(\zeta, \mathbf{A}))|$) are almost as large as the corresponding numerical radii. This fact illustrates the result from Section 5.4 that for ζ near one or more ill-conditioned eigenvalues, the numerical range of the resolvent, $(\zeta\mathbf{I} - \mathbf{A})^{-1}$, looks almost like a disk about a point close to the origin. To calculate K using Theorem 4.3, we integrate $|\lambda_{min}(\mu(\zeta(s), \mathbf{A}))|$ along the arc of the unit circle inside $W(\mathbf{A})$ and determine that $K = c_2 + \sqrt{c_2^2 + c_1} = 70.44$. Calculating K using equation (4.8) yields the smaller value $K = 36.03$. Then the K value from Theorem 4.3 is somewhat larger than the K value from (4.8), but still less than a factor of approximately 4 larger.

5.6 Remarks

K-spectral sets may be used to define helpful bounds for any norm of a function of a matrix for functions that are analytic on a subset of the complex plane containing the eigenvalues. We have demonstrated the use of K-spectral sets when bounding the convergence of iterative methods, bounding the norm of a function of a potentially block diagonal matrix, and bounding the transient growth of a dynamical system. Our extension to and numerical methods based on Theorem 4.3 allow us to place a finite upper bound on the value of K for any subset of the complex plane containing all of the eigenvalues of $\mathbf{A}$ in its interior for $f(z)$ analytic for all z in that set. Another such bound can be obtained by integrating the resolvent norm around the boundary of such a set. We proved that the integral of the resolvent norm can never be more than a factor of $4 + \epsilon$ less than the value obtained from our method based on Theorem 4.3, whereas the value obtained from Theorem 4.3 can be arbitrarily much smaller than the integral of the resolvent norm. We also partially explained where this relationship comes from.

Chapter 6

SUMMARY AND FUTURE WORK

The first part of my dissertation discussed algorithms for numerically computing the product of a function of a matrix with a vector, especially for non-Hermitian matrices. In Chapter 2, I reviewed Krylov subspace algorithms including conjugate gradient (CG), generalized minimal residual (GMRES), and bi-orthogonal conjugate gradient stabilized (Bi-CGSTAB). While CG is the most efficient algorithm in terms of computational cost and memory usage, the input matrix must be Hermitian positive definite (HPD) for CG to be guaranteed to converge.

In Section 2.2, I discussed how preconditioners can be used to further improve algorithm behavior. Algorithm 7 is an example of how preconditioners are incorporated into the CG algorithm, but they are also useful for other Krylov subspace algorithms such as GMRES and Bi-CGSTAB. Preconditioners may be used to improve the conditioning of a problem with an HPD coefficient matrix, thus improving convergence, or they may be used to turn a non-Hermitian matrix into one that is HPD.

In Section 2.2, I also introduced my work using a preconditioner to apply a CG-like algorithm to a non-Hermitian linear system where the matrix is the potential of a graph Laplacian matrix, $\mathbf{I} - \alpha\mathbf{P} = \mathbf{I} - \alpha\mathbf{D}^{-1}\mathbf{A}$, which has real valued entries. If the underlying graph is undirected, then by using the degree matrix of the graph as the preconditioner we get a real valued symmetric positive definite matrix (SPD) where CG is guaranteed to converge. Beyond the graph Laplacian example, there are many applications where the matrix of interest is not only non-Hermitian, but it is highly non-normal [108]. Not every linear system of equations containing a non-normal matrix can be transformed from the n-dimensional Hilbert space defined by the 2-norm to an appropriate n-dimensional Hilbert

space with a norm where a 3-term CG-like algorithm can be applied.

There are two conditions for when a 3-term recurrence method exists, when either the minimal polynomial of the matrix has degree three or less, or when the adjoint of the matrix $\mathbf{A}$ with respect to a specific Hilbert space is a linear function of $\mathbf{A}$ [44]. The first condition may only be satisfied by matrices with three or fewer eigenvalues. The second condition is related to why changing Hilbert spaces allows us to use a CG-like algorithm. An equivalent condition to the second one is the condition that all of the eigenvalues of the matrix lie in a straight line in the complex plane [57], for example when all of the eigenvalues of a matrix are real.

While the necessary and sufficient conditions for a 3-term recurrence have been well understood since Faber and Manteuffel's 1984 paper [44], finding the Hilbert space where a 3-term recurrence may be applied is still an open problem.

Any square matrix can be factored into a product of two symmetric matrices; however, not every matrix can be factored into a product of two Hermitian matrices [12]. While the proof of this fact relies on the Jordan normal form of a matrix, there are multiple ways to decompose many square matrices into two symmetric matrices, as is the case for the graph Laplacian potential. Another aspect of this future work is researching how other matrix decompositions can be used to form two Hermitian matrices from a single square matrix.

Also, since PCG requires that the matrix $\mathbf{A}$ be HPD or be written as the product of two HPD matrices, this algorithm cannot be used when A is indefinite. In that case, if one can factor a matrix as the product of two Hermitian matrices, one of which is positive definite, one might use a preconditioned minimum residual method (MINRES) to solve the linear system.

In Chapter 3, we introduced the optimal residual Arnoldi method for approximating the product of a matrix function with a given vector (Arnoldi-OR). Arnoldi-OR approximates $R(\mathbf{A})\mathbf{b}$, where $\mathbf{A}$ is any square matrix including a highly non-normal matrix, and $R(z)$ is a rational function with poles not equal to the eigenvalues of $\mathbf{A}$. We also introduced *a priori*

error bounds for Arnoldi-OR, including ones based on our work in K-spectral set theory discussed in Chapter 5.

One of the functions we benchmarked Arnoldi-OR against was the Arnoldi method for matrix function approximation (Arnoldi-FA), which does not currently have *a priori* error bounds except for the case where $R(z) = \frac{1}{z}$ [24]. In the case of linear systems of equations, when $R(z) = \frac{1}{z}$, there is a clear relation between the 2-norm of the Arnoldi-FA residual vector, $\mathbf{r}_k^{\mathrm{FA}}$, and the 2-norm of the Arnoldi-OR residual vector, $\mathbf{r}_k^{\mathrm{OR}}$,

$$\|\mathbf{r}_k^{\mathrm{FA}}\|_2 = \frac{\|\mathbf{r}_k^{\mathrm{OR}}\|_2}{\sqrt{1 - \left(\|\mathbf{r}_k^{\mathrm{OR}}\|_2 / \|\mathbf{r}_{k-1}^{\mathrm{OR}}\|_2\right)^2}}, \tag{6.1}$$

as stated in [23]. In the numerical examples from Section 3.3, the error of Arnoldi-FA consistently converges in the same number or fewer iterations than Arnolid-OR. If a relationship between the errors of Arnoldi-FA and Arnoldi-OR can be proven that extends to the action of any rational function of $\mathbf{A}$ on a vector, then this may be one way that general error bounds for Arnoldi-FA are proven. Meanwhile, the *a priori* error bounds for Arnoldi-OR listed in Section 3.2 can be loose overestimates of the actual convergence of Arnoldi-OR, so another research direction is to further tighten the convergence bounds of Arnoldi-OR.

Further, it would be interesting to design preconditioned versions of both the Arnoldi-FA and Arnoldi-OR algorithms. Preconditioning is a method commonly used to improve the condition number or other properties of a matrix with the goal of decreasing the number of iterations required to converge to a desired tolerance and improve the overall computational efficiency of an algorithm. For Hermitian linear systems, there are a symmetric counterparts to Arnoldi-FA and Arnoldi-OR respectively called the Lanczos method for matrix function approximation (Lanczos-FA) and the optimal residual Lanzcos method for matrix function approximation (Lanczos-OR). In the Amsel et al. paper [1], the authors used Figures 6.1 and 6.2 to show the relationship between the error from Lanczos-OR and that from Lanczos-FA, as the the condition number of the matrix $\mathbf{A}$ and degree of the denominator polynomial

$\mathbf{A}^q$ increases. These figures show that as $\kappa(\mathbf{A})$ increases, the ratio of the error output by the algorithm to the optimal error increases. For Lanczos-OR, this error ratio is bounded by a polynomial function, where the degree of the polynomial depends on q, the maximum degree of the denominator polynomial in the rational function. For Lanczos-FA, this error ratio is bounded by $\sqrt{q\kappa}$. Comparing Figures 6.2 and 6.1 from [1], Lanczos-OR is more sensitive to the condition number and maximum degree of the denominator polynomial, μ, than is Lanczos-FA. However, since the error ratio depends on $\kappa(\mathbf{A})$ for both methods, they would both benefit from preconditioning.

A good first experiment would be a similar comparison for Arnoldi-OR and Arnoldi-FA that accounts for increasing the condition number, or perhaps the level of non-normality, of $\mathbf{A}$ and increasing q, the maximum degree of the denominator polynomial. From Section 3.3.1, we already saw in Figure 3.4 that Arnoldi-OR may be more sensitive to increasing the degree of the denominator polynomial than Arnoldi-FA. Since the ratio of the 2-norm of the residual to the 2-norm of the error in Arnoldi-OR is bounded by $\kappa(D(\mathbf{A}))$, where $D(\mathbf{A})$ is the denominator polynomial, one might attempt to precondition in order to reduce the condition number of the denominator polynomial. Alternatively, since the approximate solution in Arnoldi-OR is chosen from the Krylov space $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$, one might look for a preconditioner that improves the approximation properties of the Krylov subspace. However, it is unclear how a preconditioner should be applied in the Arnoldi-OR algorithm to increase the efficiency of the algorithm in reaching a desired tolerance while balancing the increase in computational work from using a preconditioner.

Finally, in Chapter 5 we discussed extensions to K-Spectral set theory that allowed us to analytically bound the K value for a convex set equal to the numerical range with a finite number of disks removed and to numerically estimate the K value of any region containing the eigenvalues of $\mathbf{A}$ in its interior using a related numerical method based on Theorem 4.3. We also compared this numerical method to numerically calculating the bound obtained from the Cauchy integral formula, after replacing the norm of the integral by the integral

of the norm of the integrand:

$$\|f(\mathbf{A})\|_2 \leq \frac{1}{2\pi} \int_{\partial\Omega} \|(\xi\mathbf{I} - \mathbf{A})^{-1}\|_2 \, |d\xi| \sup_{z \in \partial\Omega} |f(z)|. \tag{6.2}$$

By comparing these two methods for calculating an upper bound on K for any K-spectral set, we observed some new properties of the resolvent and gave a partial explanation for these properties. We used this increased understanding to then explain some of the relationship between Theorem 4.3 and (6.2). However, a complete explanation of the relationship between Theorem 4.3 and (6.2) is still an open problem.

I also believe that a complete explanation of the relationship between Theorem 4.3 and (6.2) could be closely connected to a proof of Crouzeix's conjecture that for a square matrix $\mathbf{A}$, $\Omega = W(\mathbf{A})$ is a 2-spectral set. Overall there is still much to understand about highly non-normal matrices and numerical computations involving them.

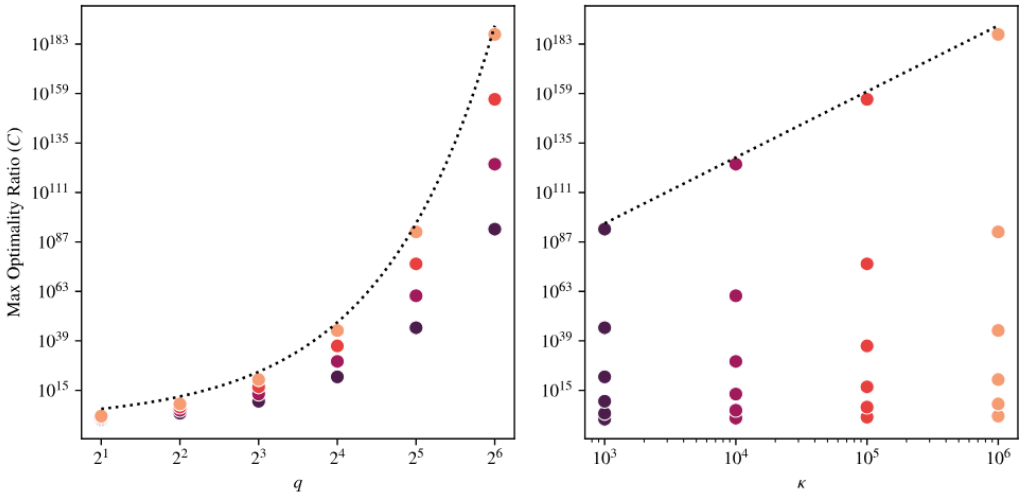


Figure 6.1: "The maximum observed ratio between the error of Lanczos-OR and the optimal error over choice of right hand side $\mathbf{b}$ when approximating $\mathbf{A}^{-q}$ for matrices with varying condition number κ. Each point shows the optimality ratio for a different pair of κ and q. Points with the same color correspond to the same value of κ. On the left, the dotted line plots $g(q) = \kappa^{q/2}$ for the maximum κ considered (10^6). On the right, the dotted line plots $g(\kappa) = \kappa^{q/2}$ for the maximum q considered ($2^6 = 64$). Overall, the optimality ratio appears to grow as $\Omega(\kappa^{q/2})$." [1]

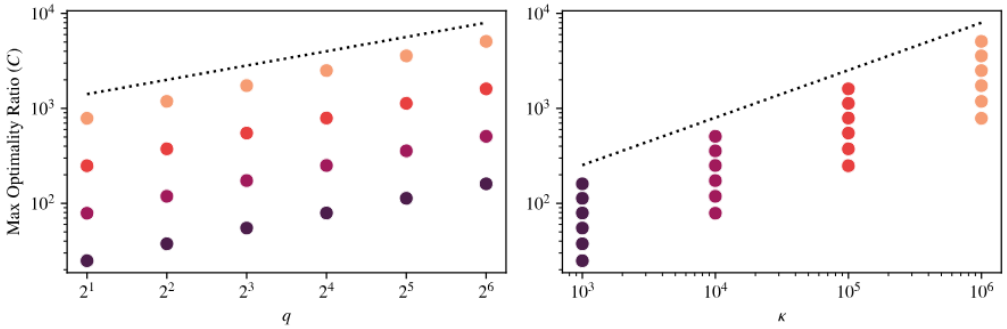


Figure 6.2: "The maximum observed ratio between the error of Lanczos-FA and the optimal error over choices of $\mathbf{b}$ when approximating $\mathbf{A}^{-q}$ for matrices with varying condition number κ. Each point corresponds to a pair (κ, q). Points with the same color have the same value of κ. On the left, the dotted line plots $\sqrt{q\kappa}$ for the maximum κ considered (10^6). On the right, the dotted line plots $\sqrt{q\kappa}$ for the maximum q considered (2^6). Overall, the optimality ratio appears to scale at least as $\Omega(\sqrt{q\kappa})$." [1]

BIBLIOGRAPHY

[1] N. Amsel, T. Chen, A. Greenbaum, C. Musco, and C. Musco, *Nearly optimal approximation of matrix functions by the Lanczos method*, Advances in Neural Information Processing Systems, 37 (2024), pp. 139823–139853.

[2] O. Balabanov and L. Grigori, *Randomized Gram–Schmidt process with application to GMRES*, SIAM Journal on Scientific Computing, 44 (2022), pp. A1450–A1474.

[3] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. M. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. Van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, Other Titles in Applied Mathematics, Society for Industrial and Applied Mathematics, Philadelphia, USA, 1994.

[4] F. L. Bauer and C. T. Fike, *Norms and exclusion theorems*, Numerische Mathematik, 2 (1960), pp. 137–141.

[5] M. Benzi, *Preconditioning techniques for large linear systems: A survey*, Journal of Computational Physics, 182 (2002), pp. 418–477.

[6] M. Benzi, *Some uses of the field of values in numerical analysis*, Bollettino dell'Unione Matematica Italiana, 14 (2021), pp. 159–177.

[7] M. Benzi and V. Simoncini, *Approximation of functions of large matrices with Kronecker structure*, Numerische Mathematik, 135 (2017), pp. 1–26.

[8] M. Berljafa, S. Elsworth, and S. Güttel, *Rational Krylov toolbox for MATLAB*.

[9] K. Bickel, P. Gorkin, A. Greenbaum, T. Ransford, F. Schwenninger, and W. E, *Crouzeix's conjecture and related problems*, Computational Methods and Function Theory, 20 (2020), pp. 1–28.

[10] R. Bollapragada, D. Scieur, and A. d'Aspremont, *Nonlinear acceleration of momentum and primal-dual algorithms*, arXiv:1810.04539 [math], (2019).

[11] D. Borba, K. S. Riedel, W. Kerner, G. T. A. Huysmans, M. Ottaviani, and P. J. Schmid, *The pseudospectrum of the resistive magnetohydrodynamics operator: Resolving the resistive Alfvén paradox*, Physics of plasmas, 1 (1994), pp. 3151–3160.

[12] A. Bosch, *The factorization of a square matrix into two symmetric matrices*, The American Mathematical Monthly, 93 (1986), pp. 462–464.

[13] M. A. Botchev, V. Grimm, and M. Hochbruck, *Residual, restarting, and Richardson iteration for the matrix exponential*, SIAM journal on scientific computing, 35 (2013), pp. A1376–A1397.

[14] M. A. Botchev, L. Knizhnerman, and M. Schweitzer, *Krylov subspace residual and restarting for certain second order differential equations*, SIAM Journal on Scientific Computing, 46 (2024), pp. S223–S253.

[15] A. Bottcher and B. Silbermann, *Introduction to Large Truncated Toeplitz Matrices*, Universitext, Springer, 1999 ed., 2012.

[16] C. Brezinski and M. Redivozaglia, *Look-ahead in Bi-CGSTAB and other product methods for linear-systems*, BIT, 35 (1995), pp. 169–201.

[17] P. N. Brown, *A theoretical comparison of the Arnoldi and GMRES algorithms*, SIAM Journal on Scientific and Statistical Computing, 12 (1991), pp. 58–78.

[18] P. N. Brown and H. F. Walker, *GMRES on (nearly) singular systems*, SIAM Journal on Matrix Analysis and Applications, 18 (1997), pp. 37–51.

[19] T. Caldwell, A. Greenbaum, and K. Li, *Some extensions of the Crouzeix–Palencia result*, SIAM Journal on Matrix Analysis and Applications, 39 (2018), pp. 769–780.

[20] F. Chaitin-Chatelin, S. for Industrial, A. Mathematics., and V. Frayssеé, *Lectures on Finite Precision Computations*, Software, environments, tools, Society for Industrial and Applied Mathematics SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104, Philadelphia, Pa, 1996.

[21] T. Chen, A. Greenbaum, C. Musco, and C. Musco, *Error bounds for Lanczos-based matrix function approximation*, SIAM Journal on Matrix Analysis and Applications, 43 (2022), pp. 787–811.

[22] ——, *Low-memory Krylov subspace methods for optimal rational matrix function approximation*, SIAM Journal on Matrix Analysis and Applications, 44 (2023), pp. 670–692.

[23] T. Chen, A. Greenbaum, and N. Wellen, *Optimal polynomial approximation to rational matrix functions using the Arnoldi algorithm*, Numerical Algorithms, (2025).

[24] T. Chen and G. Meurant, *Near-optimal convergence of the full orthogonalization method*, arXiv:2403.07259 [math.NA], (2024).

[25] D. Choi, *A proof of Crouzeix's conjecture for a class of matrices*, Linear Algebra and its Applications, 438 (2013), pp. 3247–3257.

[26] B. A. Cipra, *The best of the 20th century: Editors name top 10 algorithms*, SIAM news, 33 (2000), pp. 1–2.

[27] M. Crouzeix, *Bounds for analytical functions of matrices*, Integral equations and operator theory, 48 (2004), pp. 461–477.

[28] M. Crouzeix, *Numerical range and functional calculus in Hilbert space*, Journal of Functional Analysis, 244 (2007), pp. 668–690.

[29] M. Crouzeix and A. Greenbaum, *Spectral sets: numerical range and beyond*, SIAM journal on matrix analysis and applications, 40 (2019), pp. 1087–1101.

[30] M. Crouzeix and C. Palencia, *The numerical range is a (1+root 2)-spectral set*, SIAM journal on matrix analysis and applications, 38 (2017), p. 649–655.

[31] J. Cullum and A. Greenbaum, *Relations between Galerkin and norm-minimizing iterative methods for solving linear systems*, SIAM Journal on Matrix Analysis and Applications, 17 (1996), pp. 223–247.

[32] F. Dassi, S. Zampini, and S. Scacchi, *Robust and scalable adaptive BDDC preconditioners for virtual element discretizations of elliptic partial differential equations in mixed form*, Computer Methods in Applied Mechanics and Engineering, 391 (2022), p. 114620.

[33] B. Delyon and F. Delyon, *Generalization of von Neumann's spectral sets and integral representation of operators*, Bulletin de la Société Mathématique de France, 127 (1999), pp. 25–41.

[34] J. Demmel, *A counterexample for two conjectures about stability*, IEEE transactions on automatic control, 32 (1987), pp. 340–342.

[35] J. W. Demmel, *A numerical analyst's Jordan canonical form*, PhD thesis, University of California, Berkeley, 1983.

[36] J. Descloux, *Bounds for the spectral norm of functions of matrices*, Numerische Mathematik, 5 (1963), pp. 185–190.

[37] T. Driscoll, N. Hale, L. N. Trefethen, and eds., *Chebfun guide*, 2014.

[38] T. A. Driscoll and L. N. Trefethen, *Schwarz-Christoffel Mapping*, Cambridge Monographs on Applied and Computational Mathematics, Cambridge University Press, Cambridge, 2002.

[39] J. Drkošová, A. Greenbaum, M. Rozložník, and Z. Strakoš, *Numerical stability of GMRES*, BIT Numerical Mathematics, 35 (1995), pp. 309–330.

[40] L. Elsner and K. D. Ikramov, *Normal matrices: An update*, Linear Algebra and its Applications, 285 (1998), pp. 291–303.

[41] L. Elsner and M. H. C. Paardekooper, *On measures of nonnormality of matrices*, Linear Algebra and its Applications, 92 (1987), pp. 107–123.

[42] M. Embree, *How descriptive are GMRES convergence bounds?*, arXiv:2209.01231 [math.NA], (2022).

[43] M. Engeli, T. Ginsburg, H. Rutishauser, E. Stiefel, and E. Stiefel, *Refined Iterative Methods for Computation of the Solution and the Eigenvalues of Self-Adjoint Boundary Value Problems*, vol. 8, Springer, 1959.

[44] V. Faber and T. Manteuffel, *Necessary and sufficient conditions for the existence of a conjugate gradient method*, SIAM Journal on numerical analysis, 21 (1984), pp. 352–362.

[45] B. F. Farrell and P. J. Ionannou, *Stochastic dynamics of baroclinic waves*, Journal of the atmospheric sciences, 50 (1993), pp. 4044–4057.

[46] M. Fasi, S. Gaudreault, K. Lund, and M. Schweitzer, *Challenges in computing matrix functions*, arXiv:2401.16132 [math.NA], (2024).

[47] R. Fletcher, *Conjugate gradient methods for indefinite systems*, Lecture Notes in Mathematics, 506 (1976), pp. 73–89.

[48] A. Frommer, K. Kahl, M. Schweitzer, and M. Tsolakis, *Krylov subspace restarting for matrix Laplace transforms*, SIAM Journal on Matrix Analysis and Applications, 44 (2023), pp. 693–717.

[49] R. Gabriel, *Matrizen mit maximaler diagonale bei unitärer similarität.*, Journal für die reine und angewandte Mathematik, 307/308 (1979), pp. 31–52.

[50] D. Gaier, *Lectures on Complex Approximation*, Birkhaäuser, Boston MA, 1987.

[51] M. J. Gander and F. Nataf, *AILU: a preconditioner based on the analytic factorization of the elliptic operator*, Numerical linear algebra with applications, 7 (2000), pp. 505–526.

[52] C. Glader, M. Kurula, and M. Lindström, *Crouzeix's conjecture holds for tridiagonal 3 x 3 matrices with elliptic numerical range centered at an eigenvalue*, SIAM Journal on Matrix Analysis and Applications, 39 (2018), pp. 346–364.

[53] S. Godunov, O. Kiriljuk, and V. Kostin, *Spectral portraits of matrices*, Preprint, 3 (1990).

[54] A. Goldenberg, A. X. Zheng, S. E. Fienberg, E. M. Airoldi, et al., *A survey of statistical network models*, Foundations and Trends® in Machine Learning, 2 (2010), pp. 129–233.

[55] A. Greenbaum, *The Lanczos and conjugate gradient algorithms in finite precision arithmetic*, in Proceedings of the Cornelius Lanczos International Centenary Conference, SIAM, Philadelphia, PA, 1994, pp. 49–60.

[56] ——, *Estimating the attainable accuracy of recursively computed residual methods*, SIAM journal on matrix analysis and applications, 18 (1997), pp. 535–551.

[57] ——, *Iterative Methods for Solving Linear Systems*, Frontiers in applied mathematics, 17, Society for Industrial and Applied Mathematics SIAM, Philadelphia, PA, 1997.

[58] A. Greenbaum, F. Kyanfar, and A. Salemi, *When is the resolvent like a rank one matrix?*, arXiv:2501.07686 [math.NA], (2025).

[59] A. Greenbaum, V. Pták, and Z. Strakoš, *Any nonincreasing convergence curve is possible for GMRES*, SIAM Journal on Matrix Analysis and Applications, 17 (1996), pp. 465–469.

[60] A. Greenbaum, M. Rozložnik, and Z. Strakoš, *Numerical behaviour of the modified Gram-Schmidt GMRES implementation*, BIT Numerical Mathematics, 37 (1997), pp. 706–719.

[61] A. Greenbaum and N. Wellen, *Comparison of K-spectral set bounds on norms of functions of a matrix or operator*, Linear Algebra and its Applications, 694 (2024), pp. 52–77.

[62] R. Grone, C. R. Johnson, E. M. Sa, and H. Wolkowicz, *Normal matrices*, Linear Algebra and its Applications, 87 (1987), pp. 213–225.

[63] M. Gu, *Finding well-conditioned similarities to block-diagonalize nonsymmetric matrices is NP-hard*, Journal of Complexity, 11 (1995), pp. 377–391.

[64] S. Güttel and L. Knizhnerman, *A black-box rational Arnoldi variant for Cauchy–Stieltjes matrix functions*, BIT Numerical Mathematics, 53 (2013), pp. 595–616.

[65] W. Hackbusch, *Iterative Solution of Large Sparse Systems of Equations*, Applied Mathematical Sciences, Springer, 2 ed., 2016.

[66] G. I. Hargreaves and N. J. Higham, *Efficient algorithms for the matrix cosine and sine*, Numerical Algorithms, 40 (2005), pp. 383–400.

[67] N. Hatano and D. Nelson, *Localization transitions in non-Hermitian quantum mechanics*, Physical review letters, 77 (1996), pp. 570–573.

[68] F. Hausdorff, *Der wertvorrat einer bilinearform*, Mathematische Zeitschrift, 3 (1919), pp. 314–316.

[69] P. Henrici, *Bounds for iterates, inverses, spectral variation and fields of values of non-normal matrices*, Numerische Mathematik, 4 (1962), pp. 24–40.

[70] M. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, Journal of research of the National Bureau of Standards (1934), 49 (1952), pp. 409–.

[71] D. Hinrichsen and A. J. Pritchard, *Stability of uncertain systems*, in Systems and networks, mathematical theory and applications, U. Helmke, R. Mennicken, and J. Saurer, eds., Mathematical research, v. 77, 79, Berlin, 1994, International Symposium on the Mathematical Theory of Networks and Systems (10th : 1993 : Regensburg, Germany), Akademie Verlag.

[72] M. Hochbruck and A. Ostermann, *Exponential integrators*, Acta Numerica, 19 (2010), p. 209–286.

[73] A. J. Hoffman and H. W. Wielandt, *The variation of the spectrum of a normal matrix*, Duke Mathematical Journal, 20 (1953), pp. 37–39.

[74] R. Horn and C. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, 1991.

[75] A. S. A. S. Householder, *The Theory of Matrices in Numerical Analysis*, Introductions to higher mathematics, Blaisdell Pub. Co, New York, [1st ed.]. ed., 1964.

[76] G. F. Jonsson and L. N. Trefethen, *A numerical analyst looks at the "cutoff phenomenon" in card shuffling and other markov chains*, in Numerical Analysis, D. F. D. F. Griffiths, D. J. Higham, and G. A. G. A. Watson, eds., Pitman research notes in mathematics series, 380, Harlow, Essex, UK, 1998, Dundee Biennial Conference on Numerical Analysis (17th : 1997), Longman, pp. 150–178.

[77] W. Joubert, *On the convergence behavior of the restarted GMRES algorithm for solving nonsymmetric linear systems*, Numerical linear algebra with applications, 1 (1994), pp. 427–447.

[78] H.-O. Kreiss, *Über die stabilitätsdefinition für differenzengleichungen die partielle differentialgleichungen approximieren*, BIT, 2 (1962), pp. 153–181.

[79] R. B. Lehoucq, M. Weylandt, and J. W. Berry, *Optimal accuracy for linear sets of equations with the graph Laplacian*, arXiv:2405.07877 [math.NA], (2024).

[80] R. J. Leveque and L. N. Trefethen, *On the resolvent condition in the Kreiss matrix theorem*, BIT Numerical Mathematics, 24 (1984), pp. 584–591.

[81] Q. Liu, R. B. Morgan, and W. Wilcox, *Polynomial preconditioned GMRES and GMRES-DR*, SIAM Journal on Scientific Computing, 37 (2015), pp. S407–S428.

[82] C. Moler and C. Van Loan, *Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later*, SIAM Review, 45 (2003), pp. 3–49.

[83] M. G. Neubert and H. Caswell, *Alternatives to resilience for measuring the responses of ecological systems to perturbations*, Ecology (Durham), 78 (1997), pp. 653–665.

[84] ——, *Alternatives to resilience for measuring the responses of ecological systems to perturbations*, Ecology, 78 (1997), pp. 653–665.

[85] J. R. Norris, *Markov Chains*, Cambridge university press, 2 ed., 1998.

[86] B. O'Malley, J. Kópházi, R. P. Smedley-Stevenson, and M. D. Eaton, *P-multigrid expansion of hybrid multilevel solvers for discontinuous Galerkin finite element discrete ordinate (DG-FEM-SN) diffusion synthetic acceleration (DSA) of radiation transport algorithms*, Progress in Nuclear Energy, 98 (2017), pp. 177–186.

[87] C. C. Paige, *Accuracy and effectiveness of the Lanczos algorithm for the symmetric eigenproblem*, Linear algebra and its applications, 34 (1980), pp. 235–258.

[88] C. C. Paige and M. A. Saunders, *Solution of sparse indefinite systems of linear equations*, SIAM journal on numerical analysis, 12 (1975), pp. 617–629.

[89] S. Park, W. Lee, B. Choe, and S.-G. Lee, *A survey on personalized PageRank computation algorithms*, IEEE Access, 7 (2019), pp. 163049–163062.

[90] B. Parlett and G. Strang, *Matrices with prescribed Ritz values*, Linear Algebra and its Applications, 428 (2008), pp. 1725–1739.

[91] B. N. Parlett, *The Symmetric Eigenvalue Problem*, SIAM, 1998.

[92] S. Raouafi, *A generalization of the Kreiss matrix theorem*, Linear Algebra and its Applications, 549 (2018), pp. 86–99.

[93] A. Ruhe, *On the closeness of eigenvalues and singular values for almost normal matrices*, Linear Algebra and its Applications, 11 (1975), pp. 87–93.

[94] ——, *Closest normal matrix finally found!*, BIT Numerical Mathematics, 27 (1987), pp. 585–598.

[95] Y. Saad, *Iterative Methods for Sparse Linear Systems*, Society for Industrial and Applied Mathematics SIAM, Philadelphia, Pa, 2nd ed. ed., 2003.

[96] Y. Saad and M. H. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM Journal on Scientific and Statistical Computing, 7 (1986), pp. 856–869.

[97] A. SIEGMAN, *Lasers without photons - or should it be lasers with too many photons*, Applied physics. B, Lasers and optics, 60 (1995), pp. 247–257.

[98] P. Sonneveld, *CGS, a fast Lanczos-type solver for nonsymmetric linear systems*, SIAM journal on scientific and statistical computing, 10 (1989), pp. 36–52.

[99] B. S. Southworth, M. Holec, and T. S. Haut, *Diffusion synthetic acceleration for heterogeneous domains, compatible with voids*, Nuclear Science and Engineering, 195 (2021), pp. 119–136.

[100] M. N. Spijker, *On a conjecture by le Veque and Trefethen related to the Kreiss matrix theorem*, BIT Numerical Mathematics, 31 (1991), pp. 551–555.

[101] G. W. Stewart, *Error and perturbation bounds for subspaces associated with certain eigenvalue problems*, SIREV, 15 (1973), pp. 727–764.

[102] M. Stewart, *Perturbation of the SVD in the presence of small singular values*, Linear Algebra and its Applications, 419 (2006), pp. 53–77.

[103] Z. Strakoš, *On the real convergence rate of the conjugate gradient method*, Linear algebra and its applications, 154 (1991), pp. 535–549.

[104] J. J. Sylvester, *XIX. A demonstration of the theorem that every homogeneous quadratic polynomial is reducible by real orthogonal substitutions to the form of a sum of positive and negative squares*, The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 4 (1852), pp. 138–142.

[105] O. Toeplitz, *Das algebraische analogon zu einem satze von Fejér*, Mathematische Zeitschrift, 2 (1918), pp. 187–197.

[106] K.-C. Toh and L. N. Trefethen, *The Kreiss matrix theorem on a general complex domain*, SIAM Journal on Matrix Analysis and Applications, (2006).

[107] L. N. Trefethen, *Pseudospectra of matrices*, in Numerical analysis: Proceedings of the 14th Dundee Conference, June 1991, pp. 234–266.

[108] L. N. Trefethen, *Spectra and Pseudospectra: The Behavior of Nonnormal Matrices and Operators*, Princeton University Press, Princeton, N.J., 2005.

[109] L. N. Trefethen and D. Bau, *Numerical Linear Algebra*, SIAM, 2022.

[110] L. N. Trefethen, A. E. Trefethen, S. C. Reddy, and T. A. Driscoll, *Hydrodynamic stability without eigenvalues*, Science (American Association for the Advancement of Science), 261 (1993), pp. 578–584.

[111] A. van der Sluis, *Condition numbers and equilibration of matrices*, Numerische Mathematik, 14 (1969), pp. 14–23.

[112] H. A. van der Vorst, *Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems*, SIAM Journal on Scientific and Statistical Computing, 13 (1992), pp. 631–644.

[113] J. van Dorsselaer, J. Kraaijevanger, and M. Spijker, *Linear stability analysis in the numerical solution of initial value problems*, Acta numerica, 2 (1993), pp. 199–237.

[114] J. M. Varah, *On the separation of two matrices*, SIAM Journal on Numerical Analysis, 16 (1979), pp. 216–222.

[115] A. Vitushkin, *The analytic capacity of sets in problems of approximation theory*, Russian Mathematical Survey, 22 (1967), pp. 139–200.

[116] J. von Neumann, *Eine spektraltheorie fur allgemeine operatores lines unitaren raumes*, Math. Nachr., 4 (1951), pp. 258–281.

[117] H. F. Walker, *Implementation of the GMRES method using Householder transformations*, SIAM Journal on Scientific and Statistical Computing, 9 (1988), pp. 152–163.

[118] Wikipedia contributors, *Conjugate gradient method — Wikipedia, the free encyclopedia*, 2025. [Online; accessed 07-June-2025].

[119] J. H. Wilkinson, *Sensitivity of eigenvalues II*, Util. Math, 30 (1986), pp. 243–286.

[120] T. M. Wright and M. Embree, *EigTool: a graphical tool for nonsymmetric eigenproblems.*

[121] Y. Xi and Y. Saad, *A rational function preconditioner for indefinite sparse linear systems*, SIAM Journal on Scientific Computing, 39 (2017), pp. A1145–A1167.

[122] X. Ye, Y. Xi, and Y. Saad, *Proxy-GMRES: Preconditioning via GMRES in polynomial space*, SIAM Journal on Matrix Analysis and Applications, 42 (2021), pp. 1248–1267.

[123] Q. Zou, *GMRES algorithms over 35 years*, Applied Mathematics and Computation, 445 (2023), p. 127869.