Ruprecht-Karls-Universität Heidelberg
Faculty of mathematics and computer science
Institute of applied mathematics

Using complex shifts in Rayleigh Quotient Iteration to compute close eigenvalues

Bachelor Thesis

Author: Nils Friess

Supervisor: Prof. Dr. Robert Scheichl

Submitted: May 28, 2020

Abstract Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Zusammenfassung Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

Contents

List of Figures

List of Algorithms

Todo list

1 Introduction

This thesis we propose a novel shift-and-invert method to compute eigenvalues and eigenvectors of real symmetric matrices. The method is a modification of the *Rayleigh Quotient Iteration (RQI)*. Numerical examples suggest that our method overcomes some of the drawbacks of RQI at the expense of a slower convergence rate.

The thesis is structured as follows. In this chapter we collect some basic definitions and results from numerical linear algebra. We also review some simple iterative methods for eigenvalue problems. The second chapter is devoted to the study of the Rayleigh Quotient and Rayleigh Quotient Iteration. Among others, we give a proof for the property that makes RQI advantageous over other methods, viz., the local cubic convergence. In the third chapter we introduce our novel method that we call *Complex Rayleigh Quotient Iteration*. After motivating the method, different numerical experiments are carried out to better understand its behaviour.

1.1 The Symmetric Eigenvalue Problem

There is a plethora of examples which lead to eigenvalue problems in almost all of the natural sciences, in engineering but also other areas such as economics. In many cases the matrix of which the eigenvalues are sought is real and symmetric. The task of finding eigenvalues and eigenvectors is then referred to as the *symmetric eigenvalue problem*. For completeness, we collect some general facts from linear algebra on eigenvalues and eigenvectors below.

Definition 1.1. Let $\boldsymbol{A} \in \mathbb{C}^{n\times n}$. A scalar $\lambda \in \mathbb{C}$ is called *eigenvalue* of $\boldsymbol{A}$ if there exists a nonzero vector $\mathbf{v} \in \mathbb{C}^n$ such that

$$\boldsymbol{A}\mathbf{v} = \lambda \mathbf{v} \,. \tag{1.1}$$

The vector $\mathbf{v}$ is called an *eigenvector* of $\boldsymbol{A}$ associated with λ. The tuple $(\lambda, \mathbf{v})$ is called an *eigenpair*. The set of all eigenvalues of $\boldsymbol{A}$ is referred to as the *spectrum* and is denoted by $\sigma(\boldsymbol{A})$. To indicate that eigenvalues belong to a particular matrix $\boldsymbol{M}$ we sometimes write $\lambda(\boldsymbol{M})$.

Computing eigenpairs is a non-trivial task. Rewriting (1.1) gives $\boldsymbol{A}\mathbf{v} - \lambda\mathbf{v} = \mathbf{0}$ or $(\boldsymbol{A} - \lambda\boldsymbol{I})\mathbf{v} = \mathbf{0}$, where $\boldsymbol{I}$ is the identity matrix. Since $\mathbf{v}$ cannot be the zero vector, this equation has a solution if and only if the matrix $\boldsymbol{A} - \lambda\boldsymbol{I}$ is singular. Thus, eigenvalues of $\boldsymbol{A}$ are exactly the roots of the *characteristic polynomial*

$$\chi_{\boldsymbol{A}}(t) \coloneqq \det(\boldsymbol{A} - t\boldsymbol{I}) \,.$$

This fact, despite being of theoretical importance, cannot be used to calculate eigenvalues numerically for two reasons. First, the computation of the coefficients of the polynomial is not stable [10, p. 37]. And even if it was, it is well-known that even small perturbations in the coefficients of $\chi_{\boldsymbol{A}}(t)$ can lead to devastating errors in the roots [35, p. 190]. Thus, other methods are necessary to solve (1.1) which gave rise to iterative algorithms. These methods date back to 1846 when Jacobi published a pioneering paper on a method to compute eigenvalues of symmetric matrices [14]. Below we present essential facts from linear algebra preparing us for discussing such iterative methods in Section 1.2.

Remark 1.2 (Generalisations of eigenvalue problems)**.** The problem stated in Equation (1.1) can be generalised in multiple ways. Many problems from physics lead to the *generalised eigenvalue problem*

$$\boldsymbol{A}\mathbf{v} = \lambda \boldsymbol{M}\mathbf{v} \,. \tag{1.2}$$

In our case, we have $\boldsymbol{M} = \boldsymbol{I}$, the identity matrix. Many of the numerical algorithms for solving eigenvalue problems of the form (1.1) can be modified to solve (1.2); often certain assumptions have to be posed on $\boldsymbol{M}$ such as symmetry and positive definiteness.

Since matrices can be seen as representations of linear operators on finite-dimensional vector spaces, we can define eigenvalue problems for linear operators on more general spaces that are possibly of infinite dimension. The eigenvectors are then usually called *eigenfunctions*. Other generalisations include the *quadratic eigenvalue problem*

$$(\lambda^2 \boldsymbol{A}_2 + \lambda \boldsymbol{A}_1)\mathbf{v} = \boldsymbol{A}_0 \mathbf{v}\,,$$

with matrix coefficients $\boldsymbol{A}_0, \boldsymbol{A}_1, \boldsymbol{A}_2 \in \mathbb{C}^{n\times n}$ or more general *nonlinear eigenproblems*

$$\boldsymbol{Q}(\lambda)\mathbf{v} = \mathbf{0}\,,$$

where $\boldsymbol{Q}(\lambda)$ is a nonlinear matrix-valued function. In this thesis, we almost exclusively consider problems of the form (1.1).

In the following proposition we collect some basic facts on eigenvalues and eigenvectors. The results are shown under the assumption that $\boldsymbol{A} \in \mathbb{C}^{n\times n}$ is a complex Hermitian matrix, i. e., $\boldsymbol{A} = \boldsymbol{A}^* := \overline{\boldsymbol{A}}^{\mathsf{T}}$, where the bar denotes the complex conjugate. If $\boldsymbol{A}$ is a real matrix, we have $\overline{\boldsymbol{A}} = \boldsymbol{A}$ and thus the following facts hold in particular for real symmetric matrices.

Proposition 1.3. *Let $\boldsymbol{A} = \boldsymbol{A}^* \in \mathbb{C}^{n\times n}$ be a Hermitian matrix. Denote by $\lambda_1, \lambda_2, \ldots, \lambda_n$ the eigenvalues[1] of $\boldsymbol{A}$ with associated eigenvectors $\mathbf{v}_1, \ldots, \mathbf{v}_n$.*

(i) All eigenvalues of $\boldsymbol{A}$ are real.

[1] Of course, the eigenvalues need not be distinct. But since the eigenvalues of $\boldsymbol{A}$ are the roots of the n-degree polynomial $\chi_{\boldsymbol{A}}(t)$, when counting these roots with their multiplicity, this polynomial has n roots over $\mathbb{C}$. Thus, we can label the eigenvalues from 1 to n.

(ii) There exists an orthonormal basis of $\mathbb{C}^n$ *consisting of eigenvectors of* $\boldsymbol{A}$*. If* $\boldsymbol{A}$ *is a real symmetric matrix, the eigenvectors form an orthonormal basis of* $\mathbb{R}^n$*.*

(iii) If $\boldsymbol{A}$ *is non-singular the eigenvalues of* $\boldsymbol{A}^{-1}$ *are given by* $\lambda_1^{-1}, \dots, \lambda_n^{-1}$ *with eigenvectors* $\mathbf{v}_1, \dots, \mathbf{v}_n$*.*

(iv) Let $\mu \in \mathbb{R}$ *an arbitrary scalar. The eigenvalues of* $\boldsymbol{A} - \mu \boldsymbol{I}$ *are* $\lambda_i - \mu$ *with eigenvectors* $\mathbf{v}_1, \dots, \mathbf{v}_n$*.*

Proof. Both (i) and (ii) are well-known results from linear algebra and the proofs can be found in most standard literature (see, for example, [12, Theorem 18 and Corollary, p. 314]).

(iii) Suppose $\boldsymbol{A}$ is invertible and let $(\lambda, \mathbf{v})$ be an eigenpair of $\boldsymbol{A}$ (note that since $\boldsymbol{A}$ is non-singular we have $\lambda \neq 0$). Then

$$\boldsymbol{A}\mathbf{v} = \lambda \mathbf{v} \quad \Leftrightarrow \quad \boldsymbol{A}^{-1}\boldsymbol{A}\mathbf{v} = \lambda \boldsymbol{A}^{-1}\mathbf{v} \quad \Leftrightarrow \quad \lambda^{-1}\mathbf{v} = \boldsymbol{A}^{-1}\mathbf{v}\,,$$

hence $(\lambda^{-1}, \mathbf{v})$ is an eigenpair of $\boldsymbol{A}^{-1}$.

(iv) For $\mu \in \mathbb{R}$ and $(\lambda, \mathbf{v})$ an eigenpair we have

$$\boldsymbol{A}\mathbf{v} = \lambda \mathbf{v} \quad \Leftrightarrow \quad \boldsymbol{A}\mathbf{v} - \mu \mathbf{v} = \lambda \mathbf{v} - \mu \mathbf{v} \quad \Leftrightarrow \quad (\boldsymbol{A} - \mu \boldsymbol{I})\mathbf{v} = (\lambda - \mu)\mathbf{v}\,,$$

hence $(\lambda - \mu, \mathbf{v})$ is an eigenpair of $\boldsymbol{A} - \mu \boldsymbol{I}$.

□

In the following, we restrict our attention to the *symmetric eigenvalue problem*, i. e., we want to find solutions of Equation (1.1) assuming $\boldsymbol{A}$ is a real symmetric matrix. Thus, unless stated otherwise, for the remainder of the thesis $\boldsymbol{A}$ denotes a matrix of this type. The (real) eigenvalues are denoted by $\lambda_j(\boldsymbol{A}) = \lambda_j$ with corresponding (real) eigenvectors $\mathbf{v}_j$. Since any scalar multiple of an eigenvector is also an eigenvector, we assume that they are normalised w. r. t. the Euclidean norm so that

$$\|\mathbf{v}_i\| \coloneqq \|\mathbf{v}_i\|_2 \coloneqq \sqrt{\mathbf{v}_i^\mathsf{T}\mathbf{v}_i} = 1 \quad \text{for all } i = 1, \ldots, n\,.$$

Due to Proposition 1.3 (ii) we have

$$\langle \mathbf{v}_i, \mathbf{v}_j \rangle = \mathbf{v}_i^\mathsf{T}\mathbf{v}_j = 0 \quad \text{for } i \neq j\,,$$

where $\langle \cdot, \cdot \rangle$ denotes the Euclidean inner product on $\mathbb{R}^n$. Since all eigenvalues are real we can label them in increasing order of magnitude

$$|\lambda_1| \leq |\lambda_2| \leq \ldots \leq |\lambda_n|\,.$$

The eigenvalues λ_1 and λ_n are called *extreme* eigenvalues. The remaining eigenvalues $\lambda_2, \ldots, \lambda_{n-1}$ are called *interior* eigenvalues.

1.2 Iterative Methods for Eigenvalue Problems

With the necessary facts from linear algebra at hand we can introduce some simple iterative methods for computing eigenpairs of symmetric matrices.

We are always interested in how fast these methods produce good approximations of eigenvectors or eigenvalues (or both). The following definition provides us with a notion of the speed at which a sequence converges to its limit.

Definition 1.4 (Order of Convergence)**.** Let $(\mathbf{x}^{(k)})_{k\in\mathbb{N}}$ be a sequence in $\mathbb{C}^n$ that converges to $\mathbf{z} \in \mathbb{C}^n$.

(i) The sequence is said to converge *linearly* to $\mathbf{z}$, if there exists a constant

$0 < \rho < 1$ such that

$$\lim_{k\to\infty} \frac{\left\|\mathbf{x}^{(k+1)} - \mathbf{z}\right\|}{\left\|\mathbf{x}^{(k)} - \mathbf{z}\right\|} < \rho\,,$$

where ρ is called the *rate of convergence.*

(ii) The sequence *converges with order q to* $\mathbf{z}$ for $q > 1$ if

$$\lim_{k\to\infty} \frac{\left\|\mathbf{x}^{(k+1)} - \mathbf{z}\right\|}{\left\|\mathbf{x}^{(k)} - \mathbf{z}\right\|^q} < M\,,$$

for some $M > 0$. In particular, convergence with order

- $q = 2$ is called *quadratic convergence*,
- $q = 3$ is called *cubic convergence*

etc.

In some cases, in particular for sequences that approximate eigenvectors, the convergence behaviour is best studied in terms of the *error angle* between $\mathbf{x}^{(k)}$ and $\mathbf{z}$.

Definition 1.5 (Angle)**.** The *angle* between two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n \setminus \{\mathbf{0}\}$ is defined as

$$\angle(\mathbf{x}, \mathbf{y}) = \arccos \frac{|\langle \mathbf{x}, \mathbf{y}\rangle|}{\|\mathbf{x}\|\|\mathbf{y}\|}\,.$$

Often, the following identities are convenient

$$\sin\angle(\mathbf{x}, \mathbf{y}) := \sqrt{1 - \cos^2\angle(\mathbf{x}, \mathbf{y})}\,, \qquad \tan\angle(\mathbf{x}, \mathbf{y}) := \frac{\sin\angle(\mathbf{x}, \mathbf{y})}{\cos\angle(\mathbf{x}, \mathbf{y})}\,.$$

To see why also the angle can be used to measure the convergence speed, suppose $\mathbf{x}^{(k)}$ converges to the unit vector $\mathbf{z}$. Let $\mathbf{u}^{(k)}$ be the unit vector that lies in the span of $\mathbf{x}^{(k)}$ and $\mathbf{z}$ and is orthogonal to $\mathbf{z}$ and denote by $\phi^{(k)} = \angle(\mathbf{x}^{(k)}, \mathbf{z})$

the error angle between the current vector iterate and the limit. Now, write the vector iterate $\mathbf{x}^{(k)}$ as

$$\mathbf{x}^{(k)} = \mathbf{z}\cos\phi^{(k)} + \mathbf{u}^{(k)}\sin\phi^{(k)}\,.$$

We temporarily omit the superscripts and write $\mathbf{x} = \mathbf{x}^{(k)}$, $\mathbf{u} = \mathbf{u}^{(k)}$ and $\phi = \phi^{(k)}$. Then, using the identities $\sin^2(\phi/2) = \frac{1-\cos\phi}{2}$, $\sin^2\phi + \cos^2\phi = 1$ and the Pythagorean theorem we obtain

$$\begin{aligned}
\|\mathbf{x}-\mathbf{z}\|^2 &= \|\mathbf{z}\cos\phi + \mathbf{u}\sin\phi - \mathbf{z}\|^2 \\
&= \|\mathbf{z}(\cos\phi - 1)\|^2 + \|\mathbf{u}\sin\phi\|^2 \\
&= (\cos\phi - 1)^2 + 1 - \cos^2\phi \\
&= 2(1-\cos\phi) = 4\sin^2(\phi/2)\,.
\end{aligned}$$

Thus, convergence orders w. r. t. the norm imply the same convergence orders in terms of the error angles and vice verca.

Power method

The *power method* or *Von Mises iteration* is one of the oldest iterative methods for computing eigenvectors. It is based on generating the sequence $\mathbf{x}^{(k)} := \boldsymbol{A}^k\mathbf{x}^{(0)}$ where $\mathbf{x}^{(0)}$ is a non-zero unit vector. Of course, $\boldsymbol{A}^k$ does not have to be computed explicitly at each step since

$$\boldsymbol{A}^k\mathbf{x} = \boldsymbol{A}(\boldsymbol{A}(\ldots\boldsymbol{A}(\boldsymbol{A}\mathbf{x})\ldots))\,.$$

To prevent underflow and overflow errors, $\mathbf{x}^{(k)}$ is normalised at each step. In Algorithm 1.6 we normalise by ensuring that the largest component of the current approximation is equal to one. Of course, other norms can be used. The sequence $\mathbf{x}^{(k)}$ converges to the eigenvector associated with the eigenvalue λ_n under the assumptions that λ_n is dominant (i. e., $|\lambda_n|$ is strictly greater than $|\lambda_{n-1}|$) and that the starting vector $\mathbf{x}^{(0)}$ has a non-vanishing component in the direction of $\mathbf{v}_n$. The advantage of normalising w. r. t. the maximum norm is that the largest component of $|\boldsymbol{A}\mathbf{x}^{(k-1)}|$ converges to the eigenvalue λ_n. Regardless of the normalisation chosen, the method converges linearly with convergence rate

$$\rho = \frac{|\lambda_{n-1}|}{|\lambda_n|} \, . \tag{1.3}$$

Thus, the method can be very slow if the distance between the eigenvalues λ_n and λ_{n-1} is very small. For a detailed convergence analysis, see [31, pp. 86 sq.].

Algorithm 1.6: Power method

```
begin
    Choose nonzero initial vector $\mathbf{x}^{(0)}$
    for $k = 1, 2, \ldots$ until convergence do
        $\mathbf{x}^{(k)} = \frac{1}{\alpha^{(k)}} \boldsymbol{A}\mathbf{x}^{(k-1)}$
        /* $\alpha^{(k)}$ is the component of $\boldsymbol{A}\mathbf{x}^{(k-1)}$ with the maximum
           modulus                                                    */
```

Besides the possible slow convergence rate, the power method will always converge to an eigenvector associated with the dominant eigenvalue λ_n. In many applications, however, one already has a good approximation of another eigenvalue and wants to compute an eigenvector it belongs to. In this case, the following method can be used.

(Shifted) Inverse Iteration

The *inverse iteration* is the power method applied to $\boldsymbol{A}^{-1}$ (provided that the inverse exists). Due to Proposition 1.3 (iii) this will produce a sequence of vectors $\mathbf{x}^{(k)}$ converging to the eigenvector associated to the eigenvalue that is smallest in modulus λ_1. Combining this idea with Proposition 1.3 (iv) yields the *shifted inverse iteration*. There, the iterates are defined by

$$\mathbf{x}^{(k)} = \beta(\boldsymbol{A} - \sigma \boldsymbol{I})^{-1}\mathbf{x}^{(k-1)}\,,$$

where β is responsible for normalising $\mathbf{x}^{(k)}$. The smallest eigenvalue in modulus of the shifted matrix $\boldsymbol{A} - \sigma\boldsymbol{I}$ is the eigenvalue of $\boldsymbol{A}$ that is closest to σ. Hence, this method converges to an eigenvector associated with this eigenvalue. The most expensive step of this procedure is obviously the computation of the inverse at each step. Fortunately, this computation is not necessary since instead of explicitly computing the inverse, before the loop we can compute the LU decomposition of $\boldsymbol{A} - \sigma\boldsymbol{I}$ and solve the system $(\boldsymbol{A} - \sigma\boldsymbol{I})\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)}$ for $\mathbf{x}^{(k)}$. At each step then, only one backward and one forward substitution is required, reducing the complexity from $\mathcal{O}(n^3)$ to $\mathcal{O}(n^2)$. We summarise the results in Algorithm 1.7 (there, we normalise w. r. t. the Euclidean norm).[2]

Algorithm 1.7: Shifted inverse iteration

```
Input: Nonzero unit vector $\mathbf{x}^{(0)}$, shift $\sigma \in \mathbb{R}$
Compute $\boldsymbol{LU}$ decomposition $\boldsymbol{A} - \sigma\boldsymbol{I} = \boldsymbol{LU}$
for $k = 1, 2, \ldots$ until convergence do
    Solve $(\boldsymbol{A} - \sigma\boldsymbol{I})\tilde{\mathbf{x}}^{(k)} = \mathbf{x}^{(k-1)}$ for $\tilde{\mathbf{x}}^{(k)}$
    $\mathbf{x}^{(k)} \leftarrow \tilde{\mathbf{x}}^{(k)}/\|\tilde{\mathbf{x}}^{(k)}\|$
```

Since this is essentially the power method (applied to the inverse of $\boldsymbol{A} - \sigma\boldsymbol{I}$) this algorithm still converges linearly. However, if we denote by μ_1 the eigenvalue that is closest to the shift σ and by μ_2 the one that is the next closest one, the eigenvalue of largest modulus of $(\boldsymbol{A} - \sigma\boldsymbol{I})^{-1}$ is $1/(\mu_1 - \sigma)$ and (1.3) suggests that the convergence rate is

$$\rho = \frac{|\mu_1 - \sigma|}{|\mu_2 - \sigma|}\,.$$

[2] Note that we did not specify the "until convergence" criteria in neither of the algorithms introduced in this section. We postpone this discussion until Section 2.1.

Therefore, the method is often used to compute an eigenvector of $\boldsymbol{A}$ if a good approximation of the corresponding eigenvalue is already available.

Note, however, that a shift which is very close to an eigenvalue produces a very ill-conditioned linear system and one might expect inverse iteration to fail in this case since, in general, it is impossible to solve ill-conditioned systems accurately. Despite this seemingly obvious problem, in practice it was observed that the method produces good approximations. According to Parlett [26, pp. 84 sq.], it was Wilkinson who elucidated why the ill-conditioning is not a problem in most cases. Suppose $\sigma \approx \lambda$ where λ is an eigenvalue of $\boldsymbol{A}$ with corresponding eigenvector $\mathbf{v}$. Wilkinson illustrated that although $\tilde{\mathbf{x}}^{(k)}$ may be far from $\mathbf{v}$, the normalised solution $\mathbf{x}^{(k)} = \tilde{\mathbf{x}}^{(k)}/\|\tilde{\mathbf{x}}^{(k)}\|$ will not be far from $\mathbf{v}$ when the system is solved backwards-stably. For a detailed analysis of this problem, see [37, pp. 621–630], [26, pp. 68–71] and [28]. This will become important again later when we discuss Rayleigh Quotient Iteration. There, the system that is solved gets increasingly ill-conditioned at each step but for the same reason as above, in practice this poses no problem.

At each step in the shifted inverse iteration, better approximations for the target eigenvector are computed. There a different techniques to obtain approximations of the corresponding eigenvalue. One of these methods, which leads to the *Rayleigh Quotient Iteration*, is discussed in detail in the next chapter. For further discussion on the variants and developments of these so called *shift-and-invert* methods see, e. g., the historic survey [13] or Sections 2 and 3 of [34].

Of course there is a wide variety of other methods to compute eigenvalues and eigenvectors that are suitable for different kinds of problems. Some of them also compute only a single eigenvector (and/or eigenvalue) as the algorithms from above do, some compute the complete spectrum and full set eigenvectors while some computy only a fixed number of the eigenvalues and corresponding eigenvalues. Apart from Rayleigh Quotient Iteration and our new method called *Complex Rayleigh Quotient Iteration*, which are introduced in the second and third chapter, respectively, we do not discuss any of these other algorithms. For further discussion on the variants and developments of the different eigenvalue methods we refer to the historic survey [10] and the references therein which include the classic monographs by Saad [31] (that is concerned with large scale problems) and by Parlett [26] (that considers the symmetric eigenvalue problem).

2 Classic Rayleigh Quotient Iteration

In this chapter we introduce the Rayleigh Quotient Iteration (or RQI, for short) and discuss some important results. Among others, we give a proof of the most appealing property of RQI, namely the (local) cubic convergence.

As mentioned above, we want to find the eigenvalues and eigenvectors of a real symmetric matrix $\boldsymbol{A}$. We make the following additional assumptions:

(A1) The matrix $\boldsymbol{A}$ is large.

(A2) A good approximation of the wanted eigenvector is available.

(A3) The wanted eigenvalue lies in the interior of the spectrum.

(A4) The eigenvalues around the wanted eigenvalue are closely spaced.

The first assumption prohibits us from using an algorithm that computes the full eigendecomposition. Due to the second assumption we would like a method that makes use of the approximation in the sense that it is faster the more accurate the approximation is. Assumption (A3) seems to pose no real constraint at first sight. However, many of the different eigenvalue algorithms are only capable of finding a few eigenvalues at either end of the spectrum or they are at least considerably faster/more likely to succeed in these cases.

It will turn out that RQI is a suitable choice if we omit the last assumption. Although a different linear system has to be solved at each step, RQI often converges within a few steps when stared with a sufficiently good eigenvector approximation. The number of steps does not necessarily increase much when the eigenvalues are closely spaced. However, RQI usually fails to compute the right eigenpair in this cases and it is generally impossible to predict which eigenpair the method will converge to. The main goal of the next chapter is to design a method that overcomes this problem.

2.1 The Rayleigh Quotient

In Chapter 1 we briefly introduced some simple iterative eigenvalue methods. In essence, RQI is shifted inverse iteration where the shift is replaced by the *Rayleigh quotient* at each step.

Definition 2.1 (Rayleigh Quotient)**.** Let $\boldsymbol{A} \in \mathbb{C}^{n\times n}$. The mapping

$$\mathcal{R}_{\boldsymbol{A}} : \mathbb{C}^n \setminus \{\mathbf{0}\} \to \mathbb{C}, \qquad \mathbf{x} \mapsto \frac{\mathbf{x}^* \boldsymbol{A}\mathbf{x}}{\mathbf{x}^*\mathbf{x}}$$

is called the *Rayleigh quotient* corresponding to the matrix $\boldsymbol{A}$.[1]

We begin by discussing some basic facts.

Lemma 2.2. *Let* $\mathbf{x} \in \mathbb{C}^n \setminus \{\mathbf{0}\}$, $0 \neq \alpha, \beta \in \mathbb{C}$ *and* $\boldsymbol{A} \in \mathbb{C}^{n\times n}$.

(i) If $(\lambda, \mathbf{v})$ *is an eigenpair of* $\boldsymbol{A}$, *then* $\mathcal{R}_{\boldsymbol{A}}(\mathbf{v}) = \lambda$.

(ii) $\mathcal{R}_{\beta\boldsymbol{A}}(\alpha\mathbf{x}) = \beta\mathcal{R}_{\boldsymbol{A}}(\mathbf{x})$ *(Homogeneity)*

(iii) $\mathcal{R}_{\boldsymbol{A}-\alpha\boldsymbol{I}}(\mathbf{x}) = \mathcal{R}_{\boldsymbol{A}}(\mathbf{x}) - \alpha$ *(Translation invariance)*

Proof. (i) We can write the Rayleigh Quotient as

$$\mathcal{R}_{\boldsymbol{A}}(\mathbf{x}) = \frac{\langle \mathbf{x}, \boldsymbol{A}\mathbf{x}\rangle}{\langle \mathbf{x}, \mathbf{x}\rangle}, \tag{2.1}$$

[1] Other notations that are popular in the literature include $R_{\boldsymbol{A}}(\mathbf{x})$, $R(\boldsymbol{A}, \mathbf{x})$, $r_{\boldsymbol{A}}(\mathbf{x})$, $\sigma_{\boldsymbol{A}}(\mathbf{x})$ or $\rho_{\boldsymbol{A}}(\mathbf{x})$.

where $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^* \mathbf{y}$ denotes the Euclidean inner product on $\mathbb{C}^n$. Due to the linearity in the second argument we obtain

$$\mathcal{R}_{\boldsymbol{A}}(\mathbf{v}) = \frac{\langle \mathbf{v}, \boldsymbol{A}\mathbf{v} \rangle}{\langle \mathbf{v}, \mathbf{v} \rangle} = \frac{\langle \mathbf{v}, \lambda\mathbf{v} \rangle}{\langle \mathbf{v}, \mathbf{v} \rangle} = \lambda \frac{\langle \mathbf{v}, \mathbf{v} \rangle}{\langle \mathbf{v}, \mathbf{v} \rangle} = \lambda \,.$$

(ii) By again writing the Rayleigh Quotient as in (2.1) and using the semi-linearity in the first and linearity in the second argument of the inner product we obtain

$$\mathcal{R}_{\beta\boldsymbol{A}}(\alpha\mathbf{x}) = \frac{\langle \alpha\mathbf{x}, \beta\boldsymbol{A}(\alpha\mathbf{x}) \rangle}{\langle \alpha\mathbf{x}, \alpha\mathbf{x} \rangle} = \beta \frac{\overline{\alpha}\alpha \langle \mathbf{x}, \boldsymbol{A}\mathbf{x} \rangle}{\overline{\alpha}\alpha \langle \mathbf{x}, \mathbf{x} \rangle} = \beta \mathcal{R}_{\boldsymbol{A}}(\mathbf{x}) \,.$$

(iii)

$$\mathcal{R}_{\boldsymbol{A}-\alpha\boldsymbol{I}}(\mathbf{x}) = \frac{\mathbf{x}^*(\boldsymbol{A} - \alpha\boldsymbol{I})\mathbf{x}}{\mathbf{x}^*\mathbf{x}} = \frac{\mathbf{x}^*\boldsymbol{A}\mathbf{x} - \alpha\mathbf{x}^*\mathbf{x}}{\mathbf{x}^*\mathbf{x}} = \mathcal{R}_{\boldsymbol{A}}(\mathbf{x}) - \alpha \,.$$

□

Although the Rayleigh Quotient might look arbitrary at first sight, it occurs naturally as the solution of a least squares minimisation problem. First note that if $(\lambda, \mathbf{v})$ is an eigenpair of $\boldsymbol{A}$

$$\|\boldsymbol{A}\mathbf{v} - \lambda\mathbf{v}\| = 0 \,.$$

Now, suppose $\hat{\mathbf{v}}$ is an approximation for $\mathbf{v}$ and we want to find the best approximation $\hat{\lambda}$ for λ in the sense that

$$\hat{\lambda} = \arg\min_{\mu \in \mathbb{C}} \|\boldsymbol{A}\hat{\mathbf{v}} - \mu\hat{\mathbf{v}}\| \,.$$

This is a linear least squares problem in μ with normal equations (see, e. g. [8, Theorem 6.12, p. 362])

$$(\hat{\mathbf{v}}^*\hat{\mathbf{v}})\,\mu = \hat{\mathbf{v}}^*\boldsymbol{A}\hat{\mathbf{v}}$$

and dividing by $(\hat{\mathbf{v}}^*\hat{\mathbf{v}})$ yields the solution

$$\mu = \frac{\hat{\mathbf{v}}^* \boldsymbol{A} \hat{\mathbf{v}}}{\hat{\mathbf{v}}^* \hat{\mathbf{v}}} = \mathcal{R}_{\boldsymbol{A}}(\hat{\mathbf{v}}) \, .$$

That is, the Rayleigh Quotient is the choice of μ that minimises the *residual* norm for the eigenvalue problem. The following result specifies how good of an estimate the Rayleigh Quotient is. According to Parlett this is the property to which "the phenomenal convergence rate [of RQI] can be attributed" [26, p. 77]. We postpone the proof of this statement until when we discuss the convergence behaviour of RQI.

Lemma 2.3 (Eigenvalue estimate)**.** *Let* $\mathbf{x} \in \mathbb{C}^n$ *be an approximation of an eigenvector* $\mathbf{v}$ *of a normal*[2] *matrix* $\boldsymbol{A}$ *with corresponding eigenvalue* λ*. Then*

$$|\mathcal{R}_{\boldsymbol{A}}(\mathbf{x}) - \mathcal{R}_{\boldsymbol{A}}(\mathbf{v})| = |\mathcal{R}_{\boldsymbol{A}}(\mathbf{x}) - \lambda| = \mathcal{O}\left(\|\mathbf{x} - \mathbf{v}\|^2\right) \, .$$

This result is often paraphrased as "the Rayleigh Quotient is a *quadratically accurate* estimate of an eigenvalue" (see for example [35, p. 204]). If $\boldsymbol{A}$ is non-normal, the Rayleigh quotient is still an estimate of order one, i. e.,

$$|\mathcal{R}_{\boldsymbol{A}}(\mathbf{x}) - \mathcal{R}_{\boldsymbol{A}}(\mathbf{v})| = \mathcal{O}(\|\mathbf{x} - \mathbf{v}\|) \, .$$

We now have a method that allows us to obtain an estimation of an *eigenvalue* from an *eigenvector*. With the Shifted Inverse Iteration (Algorithm 1.7) we have the converse: a method for obtaining an *eigenvector* estimate from an *eigenvalue* estimate. Rayleigh Quotient Iteration is essentially a combination of those two methods where each step consists of one step of Shifted Inverse Iteration and the computation of the Rayleigh Quotient. We summarise the results in Algorithm 2.4. Note, that in the computation of the Rayleigh

[2] A matrix $\boldsymbol{A}$ is said to be *normal* if $\boldsymbol{A}^*\boldsymbol{A} = \boldsymbol{A}\boldsymbol{A}^*$. Note that for complex Hermitian (or real symmetric) matrices we have $\boldsymbol{A} = \boldsymbol{A}^*$, hence Hermitian (and thus symmetric) matrices are normal.

Algorithm 2.4: Rayleigh Quotient Iteration

```
Input: Nonzero unit vector $\mathbf{x}^{(0)}$
$\mu^{(0)} \leftarrow \left(\mathbf{x}^{(0)}\right)^* \boldsymbol{A}\mathbf{x}^{(0)}$
for $k = 1, 2, \ldots$ until convergence do
    Solve $\left(\boldsymbol{A} - \mu^{(k)}\boldsymbol{I}\right)\mathbf{y}^{(k)} = \mathbf{x}^{(k-1)}$ for $\mathbf{y}^{(k)}$
    $\mathbf{x}^{(k)} \leftarrow \mathbf{y}^{(k)} / \left\|\mathbf{y}^{(k)}\right\|$
    $\mu^{(k)} \leftarrow \left(\mathbf{x}^{(k)}\right)^* \boldsymbol{A}\mathbf{x}^{(k)}$
```

Quotient the division by $(\mathbf{x}^{(k)})^*\mathbf{x}^{(k)}$ can be omitted. The vector is already normalised at the previous step so that the denominator of the Rayleigh Quotient is equal to one.

We have yet to define what we mean by "until convergence" in this Algorithm but also in the simple vector iterations from Chapter 1. Now that we the defined the Rayleigh Quotient, we can define the following stopping criterion. Run the iteration until

$$\left\|\mathbf{r}^{(k)}\right\| = \left\|\boldsymbol{A}\mathbf{x}^{(k)} - \mu^{(k)}\mathbf{x}^{(k)}\right\| < \texttt{tol}\,, \tag{2.2}$$

where $\mathbf{r}^{(k)} := \boldsymbol{A}\mathbf{x}^{(k)} - \mu^{(k)}\mathbf{x}^{(k)}$ is called the *residual vector* and `tol` is a user-given error tolerance. In Algorithm 1.6 (Power method) and Algorithm 1.7 (Inverse Iteration) the Rayleigh Quotient of the current vector iterate $\mu^{(k)} = (\mathbf{x}^{(k)})^*\boldsymbol{A}\mathbf{x}^{(k)}$ is not computed. Therefore, the computation of $\mu^{(k)}$ has to be added to these algorithms.

Obviously, if for some k the tuple $(\mu^{(k)}, \mathbf{x}^{(k)})$ is an eigenpair we have $\mathbf{r}^{(k)} = \mathbf{0}$. For approximate eigenpairs we expect a small residual to imply small errors in these approximations. Details on this residual-based error control can be found in [31, Section 3.2] or [3, Section 5.2]. Here, we give only some important results without proof. A popular result, usually referred to as the *Bauer-Fike theorem* (see, e. g. [31, p. 59]) states that there exists an eigenvalue λ of $\boldsymbol{A}$

such that

$$\left|\lambda - \mu^{(k)}\right| \leq \left\|\mathbf{r}^{(k)}\right\|. \tag{2.3}$$

Thus, if the stopping criterion is fulfilled, we have $|\lambda - \mu^{(k)}| < \texttt{tol}$. For the eigenvector one can show [31, p. 63] that the following bound holds

$$\sin \angle(\mathbf{x}^{(k)}, \mathbf{v}) \leq \frac{\left\|\mathbf{r}^{(k)}\right\|}{\delta},$$

where $\mathbf{v}$ is an eigenvector associated with λ and δ is the distance from $\mu^{(k)}$ to the rest of the spectrum, i. e.,

$$\delta := \min_i \left\{|\mu^{(k)} - \lambda_i| \ : \ \lambda_i \neq \lambda\right\}.$$

Thus, if the wanted eigenvalues and its neighbours are very close, δ is large and the eigenvector might not be accurate. Due to the cubic convergence of RQI (which we will discuss below) instead of stopping the iteration as soon as the criterion in (2.2) is fulfilled, we simply run another iteration. Often, the residual norm will then already be close to working accuracy and the eigenvector is sufficiently accurate. If more information about the spectrum is available, one could also adjust the stopping criterion accordingly.

2.2 History and Recent Developments

Now that we have defined the Rayleigh Quotient and Rayleigh Quotient Iteration we give an overview of the historic developments of RQI. We also discuss recent contributions that are relevant to this thesis. Some of the results, mainly the ones that are concerned with the convergence of RQI, are discussed in more detail in Section 2.3.

2.2.1 Chronology of Rayleigh Quotient Iteration

It took about 60 years from the first mention of what is now called the Rayleigh quotient until RQI was fully defined as it is given in Algorithm 2.4. In this section we give an overview of some important milestones within this 60 years. Large parts of this overview are based on Section 4 of [34].

1894 — Lord Rayleigh In the second edition of his book titled “The Theory of Sound” John William Strutt, 3rd Baron Rayleigh [29, p. 110], proposed the following iteration for improving an approximate eigenvector $\mathbf{x}^{(0)}$:

$$\text{Solve} \quad (\boldsymbol{A} - \mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(i)})\boldsymbol{I})\,\mathbf{x}^{(i+1)} = \mathbf{e}_1\,, \tag{2.4}$$

where $\mathbf{e}_1$ denotes the first natural coordinate vector, i. e., the first column of the identity matrix and $\mathbf{x}^{(i)}$ and $\mathbf{x}^{(i+1)}$ denote the current and next iterate, respectively. In fact, Lord Rayleigh considered the generalised eigenvalue problem and so in his text, the iteration reads

$$\text{Solve} \quad (\boldsymbol{A} - \mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(i)})\boldsymbol{M})\,\mathbf{x}^{(i+1)} = \mathbf{e}_1\,.$$

1949 — Kohn In a letter to the editor Walter Kohn [15] suggests the following iteration

$$\text{Solve} \quad (\boldsymbol{A} - \mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(i)})\boldsymbol{I})\mathbf{x}^{(i+1)} = \mathbf{e}_k\,,$$

where $\mathbf{e}_k$ is *any* of the natural coordinate vectors. Without a rigorous proof Kohn argues that $\mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(i)})$ converges quadratically to an eigenvalue of $\boldsymbol{A}$ (provided that $\mathbf{x}^{(0)}$ is sufficiently close to an eigenvector of $\boldsymbol{A}$). Despite the similarity to (2.4), Kohn does not mention Lord Rayleigh’s method and it is not known whether or not he was aware of it.

1951 — Crandall In a text communicated to the Royal society of London Stephen Crandall [4] suggests

$$\text{Solve} \quad (\boldsymbol{A} - \mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(i)})\boldsymbol{I})\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)}\,. \tag{2.5}$$

Actually, Crandall also considered the generalised eigenproblem but for our comparative purposes it is sufficient to consider the case $\boldsymbol{M} = \boldsymbol{I}$. Note, that this algorithm is RQI without the normalisation step. Based on the assumption that the sequence of vectors $\mathbf{x}^{(k)}$ converges, Crandall establishes cubic convergence in this sequence and the approximate eigenvalue sequence. To see why the assumption is wrong, we assume the contrary, i. e., suppose that the sequence converges to an eigenvector, say $\mathbf{v}_k$. From (2.5) we have

$$\boldsymbol{A}\mathbf{v}_k - \mathcal{R}_{\boldsymbol{A}}(\mathbf{v}_k)\mathbf{v}_k = \mathbf{v}_k \qquad \Leftrightarrow \qquad \boldsymbol{A}\mathbf{v}_k = (1 + \mathcal{R}_{\boldsymbol{A}}(\mathbf{v}_k))\mathbf{v}_k\,,$$

and so $\mathbf{v}_k$ is an eigenvector of $\boldsymbol{A}$ with corresponding eigenvalue $1 + \mathcal{R}_{\boldsymbol{A}}(\mathbf{v}_k)$. Since we know that for any eigenvector, the value of $\mathcal{R}_{\boldsymbol{A}}(\mathbf{v}_k)$ is the eigenvalue it belongs to, this implies

$$\mathcal{R}_{\boldsymbol{A}}(\mathbf{v}_k) = 1 + \mathcal{R}_{\boldsymbol{A}}(\mathbf{v}_k)$$

which is a contradiction.

Also, Crandall establishes what is usually referred to as r-cubic convergence. This is a weaker notion of convergence than the one defined earlier, which is called q-cubic convergence (see [34, Appendix A.1] for a detailed definition of these notions of convergence).

1957 – 59 — Ostrowski Alexander Ostrowski published a series of six papers titled “On the Convergence of the Rayleigh Quotient Iteration for the Computation of the Characteristic Roots and Vectors. I-VI” [18–23]. We mention the titles here, as they represent the first mention of the term

Rayleigh Quotient Iteration.

In the first paper the author suggests the iteration

$$\text{Solve} \quad (\boldsymbol{A} - \mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(i)})\boldsymbol{I})\mathbf{x}^{(i+1)} = \boldsymbol{\eta}\,, \quad \boldsymbol{\eta} \neq \mathbf{0}\,. \tag{2.6}$$

He rigorously establishes a *quadratic* convergence rate for the sequence of Rayleigh Quotients $\mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(i)})$. He then refers to a paper of Wielandt [36] and his *fractional* or *broken iteration* (German: *gebrochene Iteration*). Inspired by Wielandt‘s method he suggests replacing the fixed vector $\boldsymbol{\eta}$ in the right hand side of (2.6) to the solution from the previous step

$$(\boldsymbol{A} - \mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(i)})\boldsymbol{I})\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)}\,, \tag{2.7}$$

starting with an arbitrary non-zero vector $\mathbf{x}^{(0)}$. He then gives a rigorous proof of the local *cubic* convergence of the sequence of Rayleigh quotients $\mu_i := \mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(i)})$, i. e.,

$$\frac{\mu_{i+1} - \lambda}{(\mu_i - \lambda)^3} \longrightarrow \gamma \quad \text{as } i \to \infty\,, \tag{2.8}$$

where λ is an eigenvalue of $\boldsymbol{A}$ and γ is a positive constant. Local convergence here means that $\mathbf{x}^{(0)}$ is assumed to be near the eigenvector corresponding to λ.

Note that (2.7) is the same algorithm previously proposed by Crandall given in Equation (2.5). Ostrowski was not aware of Crandall’s method; however, while the first paper was in press the following note was added:

> “Professor G. Forsythe has directed my attention to a paper by S. H. Crandall [. . .]. In particular, Professor Crandall establishes the *cubic character* of convergence of ξ_x in the rule (28), (29). However he does not arrive at our asymptotic formula (46), which

is the principal result of our paper.” [18, p. 241].[3]

In the beginning of the second paper [19] Ostrowski discusses this in more detail and remarks that Crandall proofed the r-cubic convergence of the sequence of eigenvalue iterates, while he showed q-cubic convergence.

More importantly, he also points out in §21 of the text that in order to assure convergence in the vector iterates (and not just the Rayleigh Quotients) one needs to normalise the vectors. With this small yet important modification of Crandall’s algorithm he fully defined RQI as it is known today.

The third paper [20] of the series addresses the non-symmetric case for which Ostrowski is also able to define a method that attains local cubic convergence. This method uses a generalised notion of the Rayleigh quotient and comes at the expense of solving two linear systems at each step instead of one. The remaining papers are also mainly concerned with the non-symmetric and non-Hermitian case and are thus not of particular interest for this thesis.

This concludes the overview of the development of RQI from Lord Rayleigh’s iteration that lacks the changing right hand side and the normalisation of the vector iterate to the complete definition of RQI by Ostrowski.

2.2.2 Further Developments and Recent Contributions

We now give an overview of the developments and contributions after the introduction of RQI in 1958. We also look at some recent contributions.

After Ostrowski’s proof of the local cubic convergence, it took another ten years until a first important result concerning the *global* convergence behaviour of RQI was presented. In 1968 Parlett and Kahan [27] showed that RQI applied to symmetric matrices converges for almost all starting vectors and in 1974 Parlett [25] generalised the result for the case when $\boldsymbol{A}$ is

[3] Here, ξ_x denotes the x-th iterate of the approximate eigenvector, i. e., in our notation $\mathbf{x}^{(x)}$. The rule (28), (29) in Ostrowski’s paper corresponds to our equation (2.7) and the asymptotic formula (46) he references is given in (2.8).

normal. A slightly more concise version of the proof has been again published in Parlett's book [26].

Of course, not only is it of interest *that* the method converges but also *which* eigenpair it converges to. The following example demonstrates that the convergence behaviour of RQI might sometimes be unexpected.

Example 2.5. This example is based on an example from [24, p. 254]. Let $\boldsymbol{A} = \operatorname{diag}(1,\ 2,\ 4)$ with eigenvalues $\lambda_1 = 1,\ \lambda_2 = 2,\ \lambda_3 = 4$ and corresponding eigenvectors $\mathbf{e}_i$, the columns of the identity matrix.

(a) First consider RQI started with the vector

$$\mathbf{x}^{(0)} = \begin{pmatrix} 0.8163392507169525 \\ -0.0004821161298470036 \\ 0.5775725022046341 \end{pmatrix}.$$

This produces the eigenpair $(1, \mathbf{e}_1)$ although

$$\mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(0)}) = 2.000770218344729\,.$$

Note, however, that computing the angle between $\mathbf{x}^{(0)}$ and the eigenvectors gives

$$\angle(\mathbf{x}^{(0)}, \mathbf{e}_1) \approx 0.61575\,, \quad \angle(\mathbf{x}^{(0)}, \mathbf{e}_2) \approx 1.57128\,, \quad \angle(\mathbf{x}^{(0)}, \mathbf{e}_3) \approx 0.95504\,,$$

that is, the angle between the initial vector and the resulting eigenvector is smallest.

(b) Now take

$$\mathbf{x}^{(0)} = (0.74278,\ 0.55709,\ 0.37139)^{\mathsf{T}}$$

the angles are

$$\angle(\mathbf{x}^{(0)}, \mathbf{e}_1) \approx 0.73358\,, \quad \angle(\mathbf{x}^{(0)}, \mathbf{e}_2) \approx 0.97992\,, \quad \angle(\mathbf{x}^{(0)}, \mathbf{e}_3) \approx 1.19029$$

> and the initial shift is $\mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(0)}) \approx 1.7241$. In this case, RQI converges to the eigenpair $(2, \mathbf{e}_2)$, i. e., not the eigenvector that has the smallest angle with $\mathbf{x}^{(0)}$ but rather the one that corresponds to the eigenvalue to which the initial shift is closest.

This example shows that there seems to be no obvious way how the computed eigenpair depends on $\mathbf{x}^{(0)}$ or $\mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(0)})$, and indeed the characterisation of the global convergence behaviour of RQI is not straightforward. In contrast, the local convergence behaviour is better understood. Ostrowski [18] defined explicit estimates of *convergence neighbourhoods* for the shift, i. e., intervals around an eigenvalue in which convergence to this eigenvalue is assured. However, these intervals depend on quantities that are not known beforehand so they are of little practical use.

Efforts have been made to obtain local convergence regions with as little knowledge about the spectrum as possible. For example, Beattie and Fox [2] derive conditions under which convergence is assured to be in a given interval assuming the number of eigenvalues contained in the interval is known. Recently, Rommes [30] derived sharper bounds for local convergence neighbourhoods. He compares RQI to a related algorithm, called the Dominant Pole Algorithm (DPA). Further, he observes that RQI does not take much advantage of the information in the initial vector $\mathbf{x}^{(0)}$. In other words, even if the initial vector is a very good approximation of the wanted eigenvector, RQI might fail. We will come back to this observation in the next chapter.

Szyld [33] suggests combining shifted inverse iteration and RQI to ensure that the computed eigenvalue lies in a given interval. He obtains criteria to switch back and forth between the two algorithms to benefit from their respective advantages (inverse iteration is guaranteed to converge to the nearest eigenvalue but convergence is merely linear; RQI possesses local cubic convergence but the global convergence behaviour is possibly erratic).

The task of identifying global convergence regions in terms of the initial vector seems to be more difficult. Parlett noted in 1980 that '[t]here appears to be no simple description of how $\mathbf{v}$ depends on $\mathbf{x}^{(0)}$" [26, p. 82] and it looks as if this statement still holds true today. Pantazis and Szyld [24] and Batterson and Smillie [1] studied *basins of attraction*, i. e., regions in the unit sphere from which RQI will converge to a specific eigenvector. They did, however, only consider the three-dimensional case.

Besides the unpredictability of the outcome of RQI another major drawback is its high cost. At every iteration a linear system has to be solved and since the system changes at each step one cannot factorise the matrix beforehand as was the case in inverse iteration. There are several obvious possibilities to reduce the cost such as to change the shift only occasionally. Sometimes, inverse iteration is run a fixed number of steps before changing the shift to the Rayleigh Quotient and Szyld's paper [33] that was mentioned above can also be interpreted as a method that reduces the computational cost for large problems. Another approach that was first studied for inverse iteration is to solve the linear system itself iteratively leading to an *inexact shift-and-invert method*. The use of iterative inner solvers in RQI (for Hermitian matrices) was studied by Simoncini and Eldén e in [32] and Notay in [17]. They both analyse how the convergence of RQI is affected by solving the linear systems only approximately. This means at each iteration an approximate solution $\mathbf{y}^{(k)}$ to

$$\left(\boldsymbol{A} - \mu^{(k)}\boldsymbol{I}\right)\mathbf{y}^{(k)} = \mathbf{x}^{(k)}$$

is sought that satisfies

$$\left\|\left(\boldsymbol{A} - \mu^{(k)}\boldsymbol{I}\right)\mathbf{y}^{(k)} - \mathbf{x}^{(k)}\right\| \leq \tau^{(k)}\,,$$

where $\tau^{(k)}$ is a tolerance that might change at each step. Also [32] shows the equivalence of inexact RQI and another iterative eigenvalue method called the *Jacobi-Davidson method.* This result is extended to the non-Hermitian case by

Freitag and Spence in [7]. They also study how different preconditioners for the linear system can be used and “tuned” to improve the outer convergence. We will briefly come back to the idea of solving the inner system inexactly when discussing our new method in the next chapter.

2.3 Convergence Analysis

We have mentioned on several occasions the local cubic convergence of RQI. In this section we give a proof of this property and we also discuss some results concerning the global convergence behaviour. We do not give a full proof of the global convergence, since parts of it are very technical and, at least for our purposes, it does not give much insight in the method.

As we have seen in the previous section, the first rigorous proof of the cubic convergence of RQI was given by Ostrowski [18] but over time a number of different, simpler proofs were presented. The one we give below follows closely the one given by Demmel [6, pp. 215 sq.].

Theorem 2.6 (Cubic convergence)**.** *Rayleigh Quotient Iteration is locally cubically convergent.*

Before giving the proof, we make some remarks. *Locally* here means that

1. it is assumed that the sequence $(\mathbf{x}^{(i)})_i$ does converge to an eigenvector of $\boldsymbol{A}$ and

2. there is a finite number of iterations for which convergence might not be cubic (the *preasymptotic phase*). For some $j \in \mathbb{N}$, however, the j-th iterate is a sufficiently good approximation of the target eigenvector, such that the sequence of subsequent iterates does converge cubically.

We defined the notion of cubic convergence in the first chapter, nonetheless we remark again here that this means the number of correct digits *triples* at each step once the error is small enough. And even if the method might not converge cubically from the beginning, in practice the preasymptotic phase rarely takes more than three steps [26, p. 77].

Some of the proofs in the literature are given under the assumption that $\boldsymbol{A}$ is diagonal. To see why no generality is lost with this assumption, consider the eigendecomposition of $\boldsymbol{A}$ and write $\boldsymbol{Q}^\mathsf{T}\boldsymbol{A}\boldsymbol{Q} = \boldsymbol{\Lambda}$, where $\boldsymbol{Q}$ is an orthogonal matrix consisting of the eigenvectors of $\boldsymbol{A}$ as its columns and $\boldsymbol{\Lambda} = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$ is the diagonal matrix of eigenvalues. We discard the superscripts for a moment and change variables in RQI[4] to $\hat{\mathbf{x}} \coloneqq \boldsymbol{Q}^\mathsf{T}\mathbf{x}$ and $\hat{\mathbf{y}} \coloneqq \boldsymbol{Q}^\mathsf{T}\mathbf{y}$. Then

$$\mu^{(i)} = \mu = \mathcal{R}_{\boldsymbol{A}}(\mathbf{x}) = \frac{\mathbf{x}^\mathsf{T}\boldsymbol{A}\mathbf{x}}{\mathbf{x}^\mathsf{T}\mathbf{x}} = \frac{\hat{\mathbf{x}}^\mathsf{T}\boldsymbol{Q}^\mathsf{T}\boldsymbol{A}\boldsymbol{Q}\hat{\mathbf{x}}}{\hat{\mathbf{x}}^\mathsf{T}\boldsymbol{Q}^\mathsf{T}\boldsymbol{Q}\hat{\mathbf{x}}} = \frac{\hat{\mathbf{x}}^\mathsf{T}\boldsymbol{\Lambda}\hat{\mathbf{x}}}{\hat{\mathbf{x}}^\mathsf{T}\hat{\mathbf{x}}} = \mathcal{R}_{\boldsymbol{\Lambda}}(\hat{\mathbf{x}}) = \mathcal{R}_{\boldsymbol{\Lambda}}(\hat{\mathbf{x}}^{(i)}) \,,$$

and $\boldsymbol{Q}\hat{\mathbf{y}}^{(i+1)} = \left(\boldsymbol{A} - \mu^{(i)}\boldsymbol{I}\right)^{-1}\boldsymbol{Q}\hat{\mathbf{x}}^{(i)}$. Hence,

$$\hat{\mathbf{y}}^{(i+1)} = \boldsymbol{Q}^\mathsf{T}\left(\boldsymbol{A} - \mu^{(i)}\boldsymbol{I}\right)^{-1}\boldsymbol{Q}\hat{\mathbf{x}}^{(i)} = \left(\boldsymbol{Q}^\mathsf{T}\boldsymbol{A}\boldsymbol{Q} - \mu^{(i)}\boldsymbol{I}\right)^{-1}\hat{\mathbf{x}}^{(i)} = \left(\boldsymbol{\Lambda} - \mu^{(i)}\boldsymbol{I}\right)^{-1}\hat{\mathbf{x}}^{(i)} \,,$$

where we used $\boldsymbol{Q}^\mathsf{T} = \boldsymbol{Q}^{-1}$, the orthogonality of $\boldsymbol{Q}$. We see that running RQI with $\boldsymbol{A}$ and $\mathbf{x}^{(0)}$ is equivalent to running RQI with $\boldsymbol{\Lambda}$ and $\hat{\mathbf{x}}^{(0)}$. Thus, we assume that $\boldsymbol{A} = \boldsymbol{\Lambda}$ is already diagonal which in particular implies that the eigenvectors of $\boldsymbol{A}$ are $\mathbf{e}_i$, the natural coordinate vectors.

Explain more detailed

Proof of Theorem 2.6. Suppose without loss of generality that $\mathbf{x}^{(i)}$ converges to $\mathbf{e}_1$. Remember that we assume that the current iterate is a sufficiently good estimate of $\mathbf{e}_1$, such that for some i we can write $\mathbf{x}^{(i)} = \mathbf{e}_1 + \mathbf{d}^{(i)}$ with $\|\mathbf{d}^{(i)}\| = \epsilon \ll 1$. To show cubic convergence, we have to verify that

$$\lim_{i\to\infty} \frac{\|\mathbf{x}^{(i+1)} - \mathbf{e}_1\|}{\|\mathbf{x}^{(i)} - \mathbf{e}_1\|^3} < M \,,$$

[4] Recall the notation from Algorithm 2.4 where $\mathbf{y} = \mathbf{y}^{(i)}$ denotes the unnormalised i-th iterate and $\mathbf{x} = \mathbf{x}^{(i)}$ denotes the same iterate after normalisation.

for some positive constant M. We know that $\|\mathbf{x}^{(i)}-\mathbf{e}_1\|^3 = \epsilon^3$, hence it suffices to show that $\|\mathbf{x}^{(i+1)} - \mathbf{e}_1\| = \mathcal{O}(\epsilon^3)$. In other words we have to show that $\mathbf{x}^{(i+1)} = \mathbf{e}_1 + \mathbf{d}^{(i+1)}$ with $\|\mathbf{d}^{(i+1)}\| = \mathcal{O}(\epsilon^3)$.

Since the vectors are normalised at each step we have

$$\begin{aligned} 1 = (\mathbf{x}^{(i)})^\mathsf{T}(\mathbf{x}^{(i)}) = (\mathbf{e}_1 + \mathbf{d}^{(i)})^\mathsf{T}(\mathbf{e}_1 + \mathbf{d}^{(i)}) &= \mathbf{e}_1^\mathsf{T}\mathbf{e}_1 + 2\mathbf{e}_1^\mathsf{T}\mathbf{d}^{(i)} + (\mathbf{d}^{(i)})^\mathsf{T}\mathbf{d}^{(i)} \\ &= 1 + 2d_1^{(i)} + \epsilon^2\,. \end{aligned}$$

where $d_1^{(i)}$ denotes the first component of the vector $\mathbf{d}^{(i)}$. Rewriting gives $d_1^{(i)} = -\epsilon^2/2$ and using the symmetry[5] of $\mathbf{\Lambda}$ we obtain

$$\begin{aligned} \mu^{(i)} &= (\mathbf{x}^{(i)})^\mathsf{T}\mathbf{\Lambda}\mathbf{x}^{(i)} = (\mathbf{e}_1 + \mathbf{d}^{(i)})^\mathsf{T}\mathbf{\Lambda}(\mathbf{e}_1 + \mathbf{d}^{(i)}) \\ &= \mathbf{e}_1^\mathsf{T}\mathbf{\Lambda}\mathbf{e}_1 + 2\mathbf{e}_1^\mathsf{T}\mathbf{\Lambda}\mathbf{d}^{(i)} + (\mathbf{d}^{(i)})^\mathsf{T}\mathbf{\Lambda}\mathbf{d}^{(i)} = \lambda_1 - \eta\,, \end{aligned}$$

where $\eta \coloneqq -2\mathbf{e}_1^\mathsf{T}\mathbf{\Lambda}\mathbf{d}^{(i)} - (\mathbf{d}^{(i)})^\mathsf{T}\mathbf{\Lambda}\mathbf{d}^{(i)} = \lambda_1\epsilon^2 - (\mathbf{d}^{(i)})^\mathsf{T}\mathbf{\Lambda}\mathbf{d}^{(i)}$. Using the fact that the spectral norm of a symmetric matrix is equal to its spectral radius, i. e., the absolute value of the largest eigenvalue $\lambda_{\max}$, we can bound $|\eta|$ as follows

$$|\eta| \leq |\lambda_1|\epsilon^2 + \|\mathbf{\Lambda}\|\|\mathbf{d}^{(i)}\|^2 \leq |\lambda_{\max}|\epsilon^2 + \|\mathbf{\Lambda}\|\epsilon^2 = 2\|\mathbf{\Lambda}\|\epsilon^2 \tag{2.9}$$

and we see that $\mu^{(i)} = \lambda_1 - \eta = \lambda_1 + \mathcal{O}(\epsilon^2)$ (note that this proves Lemma 2.3).

[5] For a symmetric matrix $\boldsymbol{B} = \boldsymbol{B}^\mathsf{T}$ holds

$$\mathbf{u}^\mathsf{T}\boldsymbol{B}\mathbf{w} = (\mathbf{u}^\mathsf{T}\boldsymbol{B}\mathbf{w})^\mathsf{T} = \mathbf{w}^\mathsf{T}(\mathbf{u}^\mathsf{T}\boldsymbol{B})^\mathsf{T} = \mathbf{w}^\mathsf{T}\boldsymbol{B}^\mathsf{T}\mathbf{u} = \mathbf{w}^\mathsf{T}\boldsymbol{B}\mathbf{u}\,.$$

We obtain

$$\begin{aligned}
\mathbf{y}^{(i+1)} &= \left(\boldsymbol{\Lambda} - \mu^{(i)}\right)^{-1}\mathbf{x}^{(i)} \\
&= \left(\frac{x_1^{(i)}}{\lambda_1 - \mu^{(i)}}, \frac{x_2^{(i)}}{\lambda_2 - \mu^{(i)}}, \dots, \frac{x_n^{(i)}}{\lambda_n - \mu^{(i)}}\right)^{\mathsf{T}} \\
&= \left(\frac{1 + d_1^{(i)}}{\lambda_1 - \mu^{(i)}}, \frac{d_2^{(i)}}{\lambda_2 - \mu^{(i)}}, \dots, \frac{d_n^{(i)}}{\lambda_n - \mu^{(i)}},\right)^{\mathsf{T}} \\
&= \left(\frac{1 - \frac{\epsilon^2}{2}}{\eta}, \frac{d_2^{(i)}}{\lambda_2 - \lambda_1 + \eta}, \dots, \frac{d_n^{(i)}}{\lambda_n - \lambda_1 + \eta},\right)^{\mathsf{T}} \\
&= \frac{1 - \frac{\epsilon^2}{2}}{\eta}\left(1, \frac{d_2^{(i)}\eta}{(1 - \frac{\epsilon^2}{2})(\lambda_2 - \lambda_1 + \eta)}, \dots, \frac{d_n^{(i)}\eta}{(1 - \frac{\epsilon^2}{2})(\lambda_n - \lambda_1 + \eta)},\right)^{\mathsf{T}} \\
&=: \frac{1 - \frac{\epsilon^2}{2}}{\eta}\left(\mathbf{e}_1 + \hat{\mathbf{d}}^{(i+1)}\right),
\end{aligned}$$

where in the first step we used that the inverse of a diagonal matrix consists of the inverse of the entries on the diagonal. The next step holds since $\mathbf{x}^{(i)} = \mathbf{e}_1 + \mathbf{d}^{(i)}$ by assumption. After that we use that we can write $\mu^{(i)} = \lambda_1 - \eta$ and $d_1^{(i)} = -\epsilon^2/2$. If we denote again by δ the gap between λ_1 and the rest of the spectrum, i. e.,

$$\delta := \min_{j=2}^{n} |\lambda_j - \lambda_1| ,$$

we can bound the denominators of $\hat{\mathbf{d}}^{(i+1)}$ using $|\lambda_j - \lambda_1 + \eta| \geq \delta - |\eta|$, and in conjunction with (2.9) we get

$$\left\|\hat{\mathbf{d}}^{(i+1)}\right\| \leq \frac{\|\mathbf{d}^{(i)}\||\eta|}{(1 - \frac{\epsilon^2}{2})(\delta - |\eta|)} \leq \frac{2\|\Lambda\|\epsilon^3}{(1 - \frac{\epsilon^2}{2})(\delta - 2\|\Lambda\|\epsilon^2)}$$

or $\|\hat{\mathbf{d}}^{(i+1)}\| = \mathcal{O}(\epsilon^3)$. Finally, since $\mathbf{x}^{(i+1)} = \mathbf{e}_1 + \mathbf{d}^{(i+1)} = \mathbf{y}^{(i+1)}/\|\mathbf{y}^{(i+1)}\|$ and

$$\frac{\mathbf{y}^{(i+1)}}{\|\mathbf{y}^{(i+1)}\|} = \frac{\frac{1-\epsilon^2/2}{\eta}\left(\mathbf{e}_1 + \hat{\mathbf{d}}^{(i+1)}\right)}{\left\|\frac{1-\epsilon^2/2}{\eta}\left(\mathbf{e}_1 + \hat{\mathbf{d}}^{(i+1)}\right)\right\|} = \frac{\mathbf{e}_1 + \hat{\mathbf{d}}^{(i+1)}}{\left\|\mathbf{e}_1 + \hat{\mathbf{d}}^{(i+1)}\right\|}$$

we see that also $\left\|\mathbf{d}^{(i+1)}\right\| = \mathcal{O}(\epsilon^3)$, which concludes the proof. □

We see that the property that the Rayleigh Quotient yields quadratically accurate approximations of eigenvalues is indeed crucial in the derivation of the cubic convergence. Some other proofs, for example the one given by Parlett [26, p. 77], proof convergence of the *error angle* $\phi^{(k)} = \angle(\mathbf{x}^{(k)}, \mathbf{v})$, i. e., he shows

$$\lim_{k\to\infty}\left|\frac{\phi^{(k+1)}}{\left(\phi^{(k)}\right)^3}\right| \leq 1\,.$$

In some texts that study inverse iteration in a more particularised fashion, cubic convergence of RQI is proofed by combining the quadratic accuracy of the Rayleigh Quotient together with convergence results from inverse iteration (since a single step of RQI is equivalent to a single step of inverse iteration with the Rayleigh quotient of the current iterate vector chosen as the shift), see for example [35, p. 208] or [3, pp. 89 sq.].

To conclude this chapter we briefly discuss some of the results concerning the global convergence behaviour of RQI. Essential for showing that RQI is globally convergent is the following fact which, according to Parlett [26, p. 85], is due to Kahan [27]. We recall that $\mathbf{r}^{(k)}$ denotes the residual at step k in RQI, i. e.,

$$\mathbf{r}^{(k)} = \left(\boldsymbol{A} - \mu^{(k)}\boldsymbol{I}\right)\mathbf{x}^{(k)}\,.$$

Lemma 2.7 (Monotonic residuals). *For every k it holds*

$$\left\|\mathbf{r}^{(k+1)}\right\| \leq \left\|\mathbf{r}^{(k)}\right\|\,.$$

Before giving the proof, we note that we can relate the k-th and $(k+1)$-st iterate of RQI by

$$\mathbf{x}^{(k)} = \beta \left(\boldsymbol{A} - \mu^{(k)} \boldsymbol{I} \right) \mathbf{x}^{(k+1)} ,$$

for some β which implies

$$\begin{aligned} \left| \left(\mathbf{x}^{(k)} \right)^* \left(\boldsymbol{A} - \mu^{(k)} \boldsymbol{I} \right) \mathbf{x}^{(k+1)} \right| &= \left| \beta^{-1} \left(\mathbf{x}^{(k)} \right)^* \mathbf{x}^{(k)} \right| \\ &= |\beta^{-1}| \left\| \mathbf{x}^{(k)} \right\| \\ &= |\beta^{-1}| \left\| \beta \left(\boldsymbol{A} - \mu^{(k)} \boldsymbol{I} \right) \mathbf{x}^{(k+1)} \right\| \end{aligned}$$

and thus

$$\left\| \left(\boldsymbol{A} - \mu^{(k)} \boldsymbol{I} \right) \mathbf{x}^{(k+1)} \right\| = \left| \left(\mathbf{x}^{(k)} \right)^* \left(\boldsymbol{A} - \mu^{(k)} \boldsymbol{I} \right) \mathbf{x}^{(k+1)} \right| . \tag{2.10}$$

Proof of Lemma 2.7.

$$\begin{aligned} &\left\| \mathbf{r}^{(k+1)} \right\| = \left\| \left(\boldsymbol{A} - \mu^{(k+1)} \boldsymbol{I} \right) \mathbf{x}^{(k+1)} \right\| \\ &\le \left\| \left(\boldsymbol{A} - \mu^{(k)} \boldsymbol{I} \right) \mathbf{x}^{(k+1)} \right\| && \text{(since } \mu^{(k+1)} \text{ minimises the residual norm)} \\ &= \left| \left(\mathbf{x}^{(k)} \right)^* \left(\boldsymbol{A} - \mu^{(k)} \boldsymbol{I} \right) \mathbf{x}^{(k+1)} \right| && \text{(by (2.10))} \\ &\le \left\| \left(\mathbf{x}^{(k)} \right)^* \left(\boldsymbol{A} - \mu^{(k)} \boldsymbol{I} \right) \right\| \left\| \mathbf{x}^{(k+1)} \right\| && \text{(by the Cauchy-Schwarz inequality)} \\ &= \left\| \left(\mathbf{x}^{(k)} \right)^* \left(\boldsymbol{A} - \mu^{(k)} \boldsymbol{I} \right) \right\| && \text{(since } \mathbf{x}^{(k+1)} \text{ is a unit vector)} \\ &= \left\| \left(\boldsymbol{A} - \mu^{(k)} \boldsymbol{I} \right) \mathbf{x}^{(k)} \right\| && \text{(since } \boldsymbol{A} \text{ is symmetric)} \\ &= \left\| \mathbf{r}^{(k)} \right\| . \end{aligned}$$

□

Theorem 2.8 (Global convergence of RQI)**.** *Let the RQI be applied to a normal matrix started with an arbitrary unit vector* $\mathbf{x}^{(0)}$*. Then, as* $k \to \infty$*, the eigenvalue sequence* $\mu^{(k)}$ *converges and either*

1. $(\mu^{(k)}, \mathbf{x}^{(k)})$ *converges to an eigenpair* $(\lambda, \mathbf{v})$*, or*

2. *the sequence* $\mu^{(k)}$ *converges to* $(\lambda_p + \lambda_q)/2$ *and* $\mathbf{x}^{(k)}$ *oscillates between vectors converging to* $\mathbf{v}_{pq}^{+}$ *and vectors converging to* $\mathbf{v}_{pq}^{-}$*, where* $\mathbf{v}_{pq}^{\pm} = (\mathbf{v}_p \pm \mathbf{v}_q)/\sqrt{2}$*, where* λ_p *and* λ_q *are eigenvalues of* $\boldsymbol{A}$ *with corresponding eigenvectors* $\mathbf{v}_p$ *and* $\mathbf{v}_q$*.*

The latter is unstable under perturbations of $\mathbf{x}^{(k)}$*.*

As mentioned above, we do not give a complete proof of this result; below we outline a rough sketch of the first statement based on the proof by Parlett in [26].

We begin by observing that the monotonicity of the residuals implies

$$\left\|\mathbf{r}^{(k)}\right\| \longrightarrow \tau \geq 0 \quad \text{as } k \to \infty \,.$$

Now, note that the sequence of vector iterates $\mathbf{x}^{(k)}$ is confined in the unit sphere, a compact subset of $\mathbb{R}^n$. One can also show that the sequence of the Rayleigh Quotients $\mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(k)})$ is confined in a compact set and thus, the sequence $\left(\mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(k)}), \mathbf{x}^{(k)}\right)_{k\in\mathbb{N}}$ has at least one limit point. Call this point $(\rho, \mathbf{z})$. This point is the limit of a subsequence $\left(\mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(j)}), \mathbf{x}^{j}\right)_{j\in\mathcal{J}}$ for some index set $\mathcal{J}$. Using the fact that the Rayleigh Quotient is a continuous function on the unit sphere we see that

$$\mathcal{R}_{\boldsymbol{A}}(\mathbf{z}) = \lim_{j} \mathcal{R}_{\boldsymbol{A}}(\mathbf{x}^{(j)}) = \rho$$

and thus

$$\|(\boldsymbol{A} - \rho)\mathbf{z}\| = \lim_{j} \left\|\mathbf{r}^{(j)}\right\| = \tau \,,$$

where the limits are for $j \to \infty$ in $\mathcal{J}$. If we assume $\tau = 0$, this shows that $(\rho, \mathbf{z})$ must be an eigenpair of $\boldsymbol{A}$. Parlett then proceeds to use the local convergence theorem to conclude that indeed $\lim_{k\to\infty} \mathbf{x}^{(k)} = \mathbf{z}$.

The second case $\tau > 0$ is much harder to analyse and does also require a number of auxiliary results relating the eigenvectors of a matrix $\boldsymbol{A}$ to its square $\boldsymbol{A}^2$. Therefore, as mentioned above, we do not discuss this part of the proof here.

Example 2.9. To illustrate the situation in the second part of the theorem consider RQI applied to the diagonal matrix $\boldsymbol{A} = \operatorname{diag}(1,\ 2,\ 4)$ with initial vector

$$\mathbf{x}^{(0)} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \end{pmatrix}^{\mathsf{T}}.$$

It is easy to check that this results in $\mu^{(k)} = 1.5$ and

$$\mathbf{x}^{(2k)} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \end{pmatrix}^{\mathsf{T}} \qquad \text{and} \qquad \mathbf{x}^{(2k+1)} = \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \end{pmatrix}^{\mathsf{T}}$$

for $k = 0, 1, 2 \ldots$. The instability mentioned in the theorem essentially prevents the method from failing in practice since roundoff errors will inevitably introduce perturbations of the vector iterate. However, this can drastically increase the number of steps before the cubic convergence takes place. In this particular example, when executed in Matlab the method needs 36 steps in total.

Matlab

Parlett states in [26, p. 61] and in [25, p. 680] that RQI converges for almost all starting vectors but Batterson and Smillie claim that this assertion is not proofed [1, p. 625]. They proceed to give a proof of this fact which is formulated in their paper as follows.

Theorem 2.10. *The set of unit vectors for which RQI does not converge to an eigenvector is a set of measure zero.*

Concluding remarks

3 Complex Rayleigh Quotient Iteration

In this chapter we introduce *Complex Rayleigh Quotient Iteration* (CRQI[1]). This is a novel shift-and-invert algorithm similar to classic RQI. Numerical experiments suggest that this new method overcomes some of the disadvantages of classic RQI. In particular, it seems to perform well in cases when there are eigenvalues very close to the target eigenvalue but a good approximation of the eigenvector is known.

3.1 Motivation

As was the case in the previous chapters we fix a real symmetric matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$. Recall that since the eigenvectors $\mathbf{v}_1, \dots, \mathbf{v}_n$ of $\boldsymbol{A}$ form an orthonormal basis of $\mathbb{R}^n$ we can write every $\mathbf{u} \in \mathbb{R}^n$ as

$$\mathbf{u} = \sum_{i=1}^{n} \alpha_i \mathbf{v}_i$$

for certain $\alpha_1, \dots, \alpha_n \in \mathbb{R}$. Suppose now that $\mathbf{u}$ is a good approximation for one of the eigenvectors, say for $\mathbf{v}_k$. Then

$$\alpha_k \approx 1 \qquad \text{and} \qquad \alpha_j \approx 0 \ \text{ for } \ j \neq k\,.$$

[1] We abbreviate the classic Rayleigh Quotient Iteration that was discussed in the previous chapter by *RQI* or *classic RQI* and the method introduced in this chapter by *CRQI*.

Due to the pairwise orthogonality of the eigenvectors this implies

$$\mathbf{u}^{\mathsf{T}}\mathbf{v}_j \approx \begin{cases} 1 & \text{if } j = k\,, \\ 0 & \text{if } j \neq k\,. \end{cases} \tag{3.1}$$

As already mentioned previously, when using classic RQI even good approximations of eigenvectors can lead to convergence to the wrong eigenpair when the gap between the target eigenvalue and eigenvalues nearby is very small. We will see in Section 3.2 that classic RQI seems to barely make use of the information incorporated in the initial vector. The main goal of complex RQI is to fix this, i. e., to alter classic RQI in such a way that for sufficiently good approximations of eigenvectors convergence to this very eigenvector is guaranteed.

To that end, the linear system that is solved at each step in RQI is perturbed in such a way that the wanted eigenvalue gets “isolated”. Of course this perturbed system will lead to wrong solutions and so we will decrease this perturbation successively until we arrive at the unperturbed problem and hope that by the time we reach the original problem, the shift is sufficiently close to the target eigenvalue.

We make use of the fact that all eigenvalues of $\boldsymbol{A}$ are real and perturb the matrix in such a way that the eigenvalues are “raised” into the complex plane. Of course, we do not want to raise them all equally but rather in such a way that the Euclidean distance between the target eigenvalue and the other eigenvalues is increased (since the close spacing of the spectrum is what makes computing the correct eigenvalue so hard in the first place). This is done by incorporating the approximation $\mathbf{u}$ of the target eigenvector $\mathbf{v}_k$ into the perturbed matrix. Let

$$\tilde{\boldsymbol{A}} \coloneqq \boldsymbol{A} - i\gamma(\boldsymbol{I} - \mathbf{u}\mathbf{u}^{\mathsf{T}})\,, \tag{3.2}$$

where $\gamma > 0$ is positive real number and i denotes the imaginary unit. Note that the matrix $\boldsymbol{I} - \mathbf{u}\mathbf{u}^\mathsf{T}$ defines the orthogonal projection onto the span of $\mathbf{u}$. Therefore, a vector $\mathbf{x}$ that is almost parallel to $\mathbf{u}$ will barely see the imaginary part $i\gamma(\boldsymbol{I} - \mathbf{u}\mathbf{u}^\mathsf{T})$ when multiplied with $\tilde{\boldsymbol{A}}$ and so $\tilde{\boldsymbol{A}}\mathbf{x} \approx \boldsymbol{A}\mathbf{x}$. If, however, the vector $\mathbf{x}$ is almost perpendicular to $\mathbf{u}$ we have

$$\tilde{\boldsymbol{A}}\mathbf{x} = \boldsymbol{A}\mathbf{x} - i\gamma\mathbf{x} - \mathbf{u}\underbrace{\mathbf{u}^\mathsf{T}\mathbf{x}}_{\approx 0} \approx (\boldsymbol{A} - i\gamma\boldsymbol{I})\mathbf{x}\,. \tag{3.3}$$

Since $\mathbf{u}$ approximates $\mathbf{v}_k$, the orthogonal complement of the span of $\mathbf{u}$ approximates the orthogonal complement of the span of $\mathbf{v}_k$. Since the latter is the subspace spanned by the remaining eigenvectors, we expect that the eigenvectors of $\tilde{\boldsymbol{A}}$ are similar to those of $\boldsymbol{A}$ and that the eigenvalues corresponding to eigenvectors $\mathbf{v}_j$, $j \neq k$ to approximately be $\lambda_j - i\gamma$. The eigenvalue corresponding to $\mathbf{v}_k$ would then be approximately equal to λ_k since $\tilde{\boldsymbol{A}}\mathbf{v}_k \approx \boldsymbol{A}\mathbf{v}_k = \lambda_k\mathbf{v}_k$.

To make this intuition more quantitative, we decompose $\tilde{\boldsymbol{A}}$ into the sum $\tilde{\boldsymbol{A}} = \tilde{\boldsymbol{A}}_{(0)} + \tilde{\boldsymbol{A}}_{(1)}$, where

$$\tilde{\boldsymbol{A}}_{(0)} \coloneqq \boldsymbol{A} - i\gamma(\boldsymbol{I} - \mathbf{v}_k\mathbf{v}_k^\mathsf{T}) \tag{3.4}$$

and

$$\tilde{\boldsymbol{A}}_{(1)} \coloneqq i\gamma(\mathbf{u}\mathbf{u}^\mathsf{T} - \mathbf{v}_k\mathbf{v}_k^\mathsf{T}) \tag{3.5}$$

and first analyse the individual summands. The proofs for the following results can be found in the Appendix.

Lemma 3.1. *The matrix $\tilde{\boldsymbol{A}}_{(0)}$ has the same eigenvectors as $\boldsymbol{A}$ with corresponding eigenvalues $\lambda_j(\tilde{\boldsymbol{A}}_{(0)}) = \lambda_j(\boldsymbol{A}) - i\gamma$ for $j \neq k$ and $\lambda_k(\tilde{\boldsymbol{A}}_{(0)}) = \lambda_k(\boldsymbol{A})$.*

Since $\mathbf{u}$ approximates $\mathbf{v}_k$ we expect the effect on the eigenvalues of $\tilde{\boldsymbol{A}}_{(0)}$ due to the perturbation $\tilde{\boldsymbol{A}}_{(1)}$ to be small. The following lemma answers how “big” this perturbation is.

Rewrite

Lemma 3.2. *Let $\tilde{\boldsymbol{A}}_{(1)}$ be the matrix defined in Equation* (3.5)*. Then*

$$\left\|\tilde{\boldsymbol{A}}_{(1)}\right\| = \gamma\sqrt{1 - \langle \mathbf{u}, \mathbf{v}_k \rangle^2}\,, \tag{3.6}$$

where $\|\cdot\|$ denotes the spectral norm of a matrix.

For good approximations $\mathbf{u}$ we have $\langle \mathbf{u}, \mathbf{v}_k \rangle^2 \approx 1$ and thus, if γ is sufficiently small, $\|\tilde{\boldsymbol{A}}_{(1)}\| \ll 1$. We can give an estimate of the eigenvalues of $\tilde{\boldsymbol{A}}$.

Proposition 3.3. *The eigenvalues of $\tilde{\boldsymbol{A}}$ satisfy*

$$\lambda_j(\tilde{\boldsymbol{A}}) = \lambda_j(\boldsymbol{A}) + i\gamma\left(\langle \mathbf{v}_j, \mathbf{u} \rangle^2 - 1\right) + \mathcal{O}\left(\left\|\tilde{\boldsymbol{A}}_{(1)}\right\|^2\right)\,. \tag{3.7}$$

To see why this estimate matches our intuitive arguments from above we consider the middle part of (3.7). For $j \neq k$ the scalar product is approximately zero and thus, if we ignore the last term of the sum (which is small according to the previous lemma), we get

$$\lambda_j(\tilde{\boldsymbol{A}}) \approx \lambda_j(\boldsymbol{A}) - i\gamma\,.$$

For $j = k$ the middle part is small due to the scalar product being approximately one and so $\lambda_k(\tilde{\boldsymbol{A}}) \approx \lambda_k(\boldsymbol{A})$. In summary, the eigenvalue corresponding to the wanted eigenvector stays near the real line whereas the remaining eigenvalues are raised into the lower complex half-plane.

If we would use this matrix in RQI, the results would of course not be the target eigenpair but an eigenpair of $\tilde{\boldsymbol{A}}$. Thus, instead of keeping this matrix fixed, we replace the vector $\mathbf{u}$ by the current iterate $\mathbf{x}^{(k)}$ and the scalar γ by a sequence $\gamma^{(k)}$ that converges to zero. Ideally, in the beginning of the iteration $\gamma^{(k)}$ should be sufficiently large such that the target eigenvalue is properly isolated. As the iterates get closer to the target eigenpair, $\gamma^{(k)}$ should decrease such that in the end $\tilde{\boldsymbol{A}} \approx \boldsymbol{A}$. A possible choice that comes to mind is the norm of the current residual $\mathbf{r}^{(k)}$ or related quantities such as the square of the residual norm. How the choice of this shift influences the convergence behaviour is discussed in Section 3.2.

To summarise the idea of CRQI we consider one step of the iteration of classic RQI. The system that has to be solved is now of the form

$$\left(\boldsymbol{A} - i\gamma^{(k)}\left(\boldsymbol{I} - \mathbf{x}^{(k)}(\mathbf{x}^{(k)})^{*}\right) - \mu^{(k)}\boldsymbol{I}\right)\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)}\,. \tag{3.8}$$

As $\gamma^{(k)} \to 0$ this becomes the same linear system as in classic RQI. The remaining steps are essentially the same except that every transpose (e. g. in the computation of the Rayleigh Quotient) has to be replaced by a complex conjugate transpose.

Obviously, we cannot expect this method to possess the same convergence properties as classic RQI. The matrix $\tilde{\boldsymbol{A}}$ is neither real symmetric nor Hermitian but *complex symmetric*. As we will see shortly, however, if good approximations of the wanted eigenvector are used for the initial vector, the method often requires merely three or four additional steps compared to classic RQI.

Another problem seems to be that if $\boldsymbol{A}$ is sparse, $\tilde{\boldsymbol{A}}$ will in general not be sparse. In fact, the perturbed matrix will quite possibly posses only few, if any, zero entries. Although this might seem like a serious disadvantage it will shortly be resolved. We will see that it is not necessary to perturb the matrix by $-i\gamma(\boldsymbol{I} - \mathbf{u}\mathbf{u}^{\mathsf{T}})$ but that it suffices to subtract from $\boldsymbol{A}$ the diagonal matrix $i\gamma\boldsymbol{I}$. The way that we introduced the method above was merely for motivational purposes. The resulting matrix is still complex symmetric but now only the diagonal entries are altered and sparsity is preserved. We begin by showing the following result.

Lemma 3.4. *Define the matrices*

$$\boldsymbol{B} \coloneqq \boldsymbol{A} - \mu\boldsymbol{I} - i\gamma\boldsymbol{I}$$

and

$$\boldsymbol{C} := \boldsymbol{A} - i\gamma(\boldsymbol{I} - \mathbf{u}\mathbf{u}^*) - \mu\boldsymbol{I}\,,$$

where $\mu, \gamma > 0$ are positive real numbers. Then $\boldsymbol{B}^{-1}\mathbf{u}$ is a scalar multiple of $\boldsymbol{C}^{-1}\mathbf{u}$.

Proof. Let $\hat{\boldsymbol{A}} := \boldsymbol{A} - \mu\boldsymbol{I}$ and rewrite $\boldsymbol{B}$ and $\boldsymbol{C}$ as

$$\boldsymbol{B} = \hat{\boldsymbol{A}} - i\gamma\boldsymbol{I} \qquad \text{and} \qquad \boldsymbol{C} = \hat{\boldsymbol{A}} - i\gamma(\boldsymbol{I} - \mathbf{u}\mathbf{u}^*)\,,$$

respectively. Now observe that $\boldsymbol{C} = \boldsymbol{B} + i\gamma\mathbf{u}\mathbf{u}^*$. Using the Sherman-Morrison formula ([11, p. 50]) and letting $\alpha := 1 + i\gamma\mathbf{u}^*\boldsymbol{B}^{-1}\mathbf{u} \in \mathbb{C}$ we obtain

$$\begin{aligned}
\boldsymbol{C}^{-1}\mathbf{u} &= (\boldsymbol{B} + i\gamma\mathbf{u}\mathbf{u}^*)^{-1}\mathbf{u} \\
&= \left(\boldsymbol{B}^{-1} - \alpha^{-1}\boldsymbol{B}^{-1}\mathbf{u}i\gamma\mathbf{u}^*\boldsymbol{B}^{-1}\right)\mathbf{u} \\
&= \boldsymbol{B}^{-1}\mathbf{u} - \boldsymbol{B}^{-1}\mathbf{u}\,\underbrace{\alpha^{-1}i\gamma\mathbf{u}^*\boldsymbol{B}^{-1}\mathbf{u}}_{\in\mathbb{C}} \\
&= \boldsymbol{B}^{-1}\mathbf{u}(1 - \alpha^{-1}i\gamma\mathbf{u}^*\boldsymbol{B}^{-1}\mathbf{u})\,.
\end{aligned}$$

Thus, the vector $\boldsymbol{C}^{-1}\mathbf{u}$ is a scalar multiple of $\boldsymbol{B}^{-1}\mathbf{u}$. □

This result shows that it suffices to perturb the matrix by the diagonal matrix $\gamma^{(k)}i\boldsymbol{I}$ instead of $i\gamma^{(k)}(\boldsymbol{I} - \mathbf{x}^{(k)}(\mathbf{x}^{(k)})^*)$. Consider the linear system from (3.8). The lemma states that there exists a scalar $\alpha \in C$ such that (3.8) is equivalent to

$$\alpha\left(\boldsymbol{A} - i\gamma^{(k)}\boldsymbol{I} - \mu^{(k)}\boldsymbol{I}\right)\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)}\,. \tag{3.9}$$

Hence, the solution of (3.9) *without* the factor α is a scalar multiple of the solution from (3.8). Since the result gets normalised immediately after, both solutions will result in the same vector that is used in the next iteration. The Rayleigh Quotient that is computed using this vector consequently is the same in either of the variants.

Thus, it does indeed suffice to perturb the matrix only by the diagonal matrix $-i\gamma^{(k)}\boldsymbol{I}$. We can collect the real shift $\mu^{(k)}$ and the imaginary shift $\gamma^{(k)}i$ into a complex-valued shift

$$\sigma^{(k)} := \mu^{(k)} + i\gamma^{(k)}$$

and can define Complex Rayleigh Quotient Iteration as it is given in Algorithm 3.5.

Algorithm 3.5: Complex Rayleigh Quotient Iteration

```
Input: Nonzero initial vector $\mathbf{x}^{(0)}$ with $\|\mathbf{x}^{(0)}\| = 1$
begin
    $\mu^{(0)} \leftarrow (\mathbf{x}^{(0)})^{\mathsf{T}}\boldsymbol{A}\mathbf{x}^{(0)}$
    $\gamma^{(0)} \leftarrow \|(\boldsymbol{A} - \mu^{(0)}\boldsymbol{I})\mathbf{x}^{(0)}\|$
    $\sigma^{(0)} \leftarrow \mu^{(0)} + i\gamma^{(0)}$
    for $k = 1, 2, \dots$ until convergence do
        Solve $(\boldsymbol{A} - \sigma^{(k)}\boldsymbol{I})\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)}$
        $\mathbf{x}^{(k+1)} \leftarrow \tilde{\mathbf{x}}^{(k+1)}/\|\tilde{\mathbf{x}}^{(k+1)}\|$
        $\mu^{(k+1)} \leftarrow (\mathbf{x}^{(k+1)})^{*}\boldsymbol{A}\mathbf{x}^{(k+1)}$
        $\gamma^{(k+1)} \leftarrow \|(\boldsymbol{A} - \mu^{(k+1)}\boldsymbol{I})\mathbf{x}^{(k+1)}\|$
        $\sigma^{(k+1)} \leftarrow \mu^{(k+1)} + i\gamma^{(k+1)}$
    $\mathbf{x} \leftarrow \Re(\mathbf{x}^{(k+1)})$
    $\mathbf{x} \leftarrow \mathbf{x}/\|\mathbf{x}\|$
    $\mu \leftarrow \mathbf{x}^{\mathsf{T}}\boldsymbol{A}\mathbf{x}$
```

The last three steps after the loop remove the imaginary part that might be “left” in the vector approximation ($\Re(\cdot)$ extracts the real part of a complex vector). Since we again use a stopping criterion based on the norm of the residual, the same error bounds we discussed earlier in Chapter 2 hold and thus the vector iterates could still contain a small imaginary part. This does also imply that the final residual norm $\|\boldsymbol{A}\mathbf{x} - \mu\mathbf{x}\|$ will be larger than the residual norm computed in the last iteration.

Finish this part

Note, that although the step where the linear system is solved looks almost identical to the respective step in classic RQI, this algorithm is still different. The shift $\sigma^{(k)}$ is not the Rayleigh Quotient of the vector iterate $\mathbf{x}^{(k)}$ as is the case in classic RQI. One could try to interpret the method as classic RQI applied to the matrix $\boldsymbol{A} - i\gamma^{(k)}\boldsymbol{I}$ but this is also not true since this matrix changes from one step to the next.

Hence, we can obviously not expect the convergence results of classic RQI to straightforwardly carry over.

Conclude motivations

3.2 Numerical Experiments

In this section we discuss different numerical examples to better understand the behaviour of CRQI. Throughout the section we will always compare the method to classic RQI. All experiments were executed in Matlab 9 [16]. The criterion for convergence was $\|\mathbf{r}^{(k)}\| = \|\boldsymbol{A}\mathbf{x}^{(k)} - \mu^{(k)}\mathbf{x}^{(k)}\| < 10^{-9}$. The source codes for both classic and complex RQI can be found in the Appendix. Note, that he methods are defined slightly differently from the Algorithms given above. They both allow to explicitly set the initial shifts $\mu^{(0)}$ and $\gamma^{(0)}$ to specific values whereas in Algorithm 2.4 and Algorithm 3.5 the initial shifts are always initialised as the Rayleigh Quotient of the initial vector and the initial residual, respectively. This will become useful when investigating how the initial shift influences the outcome of the method.

All of the experiments were carried out for different matrices that are common for assessing eigenvalue algorithms. In particular, they have been used to evaluate shift-and-invert type algorithms and algorithms related to classic RQI, see for example [5] and the references therein. They are described below. Recall that we are particularly interested in problems with closely spaced eigenvalues and, except for the first two examples, all of the test matrices will lead to such problems. Since in three of the test problems the spacing even becomes closer the larger the matrix gets, we will also study the algorithm's behaviour under increasing matrix size.

Random matrices These matrices are pseudo-randomly generated using the Matlab command "sprandsym". Apart from being sparse and symmetric, they have no special structure or properties.

The matrix $[1, 2, 1]$ This matrix is a tridiagonal matrix with all diagonal elements set to 2 and all off-diagonal elements set to 1. The eigenvalues of this matrix are all distinct and their exact values are known to be

$$\lambda_k = 4\sin^2\left(\frac{\pi k}{2(n+1)}\right)$$

where n denotes the order (i. e., the number of rows) of the matrix [37, pp. 299–307]. Thus, for every n, the eigenvalues lie in the interval $[0, 4]$ and so consequently their spacing gets smaller the larger the matrix gets.

Wilkinson's matrix $\boldsymbol{W}^+$ This tridiagonal matrix was extensively studied by Wilkinson in [37], hence its name. It is of order $n = 2p + 1$ for some $p \in \mathbb{N}$. The i-th diagonal entry is set to $|p + 1 - i|$ and all off-diagonal entries are set to 1. For instance, for $p = 2$ this yields the 5×5 matrix

$$\boldsymbol{W}_5^+ = \begin{bmatrix} 2 & 1 & & & \\ 1 & 1 & 1 & & \\ & 1 & 0 & 1 & \\ & & 1 & 1 & 1 \\ & & & 1 & 2 \end{bmatrix}.$$

The spectrum consists of pairs of nearly, but not exactly, equal eigenvalues. For example, the two largest eigenvalues of $\boldsymbol{W}_{21}^+$ agree to 15 significant decimals. [37, pp. 300–309]

The Martin-Wilkinson matrix $\boldsymbol{MW}$ If we denote by a_{ij} the j-th entry of the i-th row of $\boldsymbol{MW}$, this matrix is defined by

$$\begin{aligned} a_{ii} &= 6 && \text{for } i = 2, \dots, n-1 \quad \text{but } a_{11} = a_{nn} = 5, \\ a_{i,i-1} = a_{i-1,i} &= -4 && \text{for } i = 2, \dots, n, \\ a_{i,i-2} = a_{i-2,i} &= 1 && \text{for } i = 2, \dots, n\,. \end{aligned}$$

The eigenvalues are known to be [9, p. 39]

$$\lambda_k = 16 \sin^4 \left(\frac{k\pi}{2(n+1)} \right)$$

and thus, independent of n, the eigenvalues lie in the interval $[0, 16]$. Hence, the eigenvalues become increasingly close the larger the size of the matrix.

Laplace matrix $\boldsymbol{L}$ This matrix arises, for instance, when discretizing the Laplace operator using a five-point stencil in the finite difference method (see, for example, [6, pp. 270–272]). Let m be an integer and define the $m \times m$ matrix

$$\boldsymbol{T} = \begin{bmatrix} 4 & -1 & & & \\ -1 & 4 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 4 & -1 \\ & & & -1 & 4 \end{bmatrix}.$$

The Laplace matrix is a block tridiagonal matrix of order $n = m^2$ defined as

$$\boldsymbol{L} = \begin{bmatrix} \boldsymbol{T} & -\boldsymbol{I} & & & \\ -\boldsymbol{I} & \boldsymbol{T} & -\boldsymbol{I} & & \\ & \ddots & \ddots & \ddots & \\ & & -\boldsymbol{I} & \boldsymbol{T} & -\boldsymbol{I} \\ & & & -\boldsymbol{I} & \boldsymbol{T} \end{bmatrix}$$

where $\boldsymbol{I}$ is the $m \times m$ identity matrix. Again, the eigenvalues can be computed analytically and all lie in an interval that is independent of the matrix's size; in this case they are all contained in the interval $[0, 8]$.

Below we give some numerical examples that highlight different behavioural aspects of the algorithm. Plots and/or tables that illustrate our findings are given for each of the test matrices. They are either included directly within the example or can be found in the appendix. Those included in this section are the ones we found most meaningful for the particular example.

Recall that we assume that a good approximation of the wanted is available. To simulate this, the full set of eigenvectors is computed using built-in Matlab routines. These vectors are then collected as columns of a matrix $\boldsymbol{V}$. Next, a weight vector $\mathbf{w}$ of uniformly distributed numbers between 0 and 1 is created. One of the components is set to a higher value than the others, e. g. $w_{50} = 10$ (in most examples the index of the component was also chosen randomly). The initial vector is then set to a weighted linear combination of the eigenvectors, i. e., $\mathbf{x}^{(0)} = \beta \boldsymbol{V} \mathbf{w}$, where β has to be chosen such that $\mathbf{x}^{(0)}$ is normalised. Now, $\mathbf{x}^{(0)}$ is a vector with a strong component in the direction of the target eigenvector (in this case this would be $\mathbf{v}_{50}$) and random (smaller) contributions in the directions of the other eigenvectors.

The first example demonstrates how difference choices for the imaginary shift $i\gamma^{(k)}$ affect the convergence speed. We run three different variants of CRQI: The first uses $\gamma^{(k)} = \|\mathbf{r}^{(k)}\|$, which is the same as in Algorithm 3.5

above. The second uses the square of the residual norm, the third approach is explained below. A plot of the residuals norm and the iteration number for all three approaches together with the results using classic RQI[2] is given in Figure 3.1.

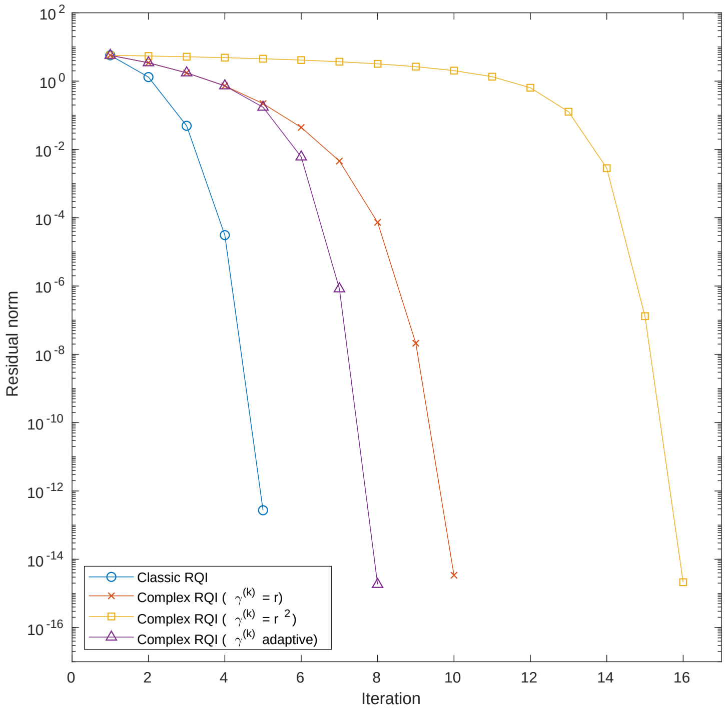


Figure 3.1: Plot of residuals of Classic RQI, Complex RQI with $\gamma^{(k)} = \|\mathbf{r}^{(k)}\|$, Complex RQI with $\gamma^{(k)} = \|\mathbf{r}^{(k)}\|^2$ and Complex RQI with the imaginary shift chosen adaptively (see text) using a random matrix. The last variant seems to combine the advantages of the second and third alternatives.

We observe that in the initial phase of the iteration, the second variant (squared residual norm) seems to be slower than the first. Although it might not be clearly observable in the figure, further investigation of other examples

[2] Classic RQI is merely included for speed comparison. In most of the examples it failed to converge to the correct eigenvalue.

suggested that during the final steps of the iterations the second version was faster than the first (see also Figures B.1 to B.4 in the Appendix). Consequently, by combining both approaches and thus changing the shift adaptively we expect faster convergence. The results are also plotted in Figure 3.1 (labeled with "$\gamma^{(k)}$ adaptive") and are in accordance with the expectation. In this particular case, the shift was changed according to

$$\gamma^{(k)} = \begin{cases} \|\mathbf{r}^{(k)}\| & \text{if } \|\mathbf{r}^{(k)}\| \geq 1\,, \\ \|\mathbf{r}^{(k)}\|^2 & \text{if } \|\mathbf{r}^{(k)}\| < 1\,. \end{cases}$$

In the following, when we speak of CRQI we mean CRQI performed with this adaptive choice of imaginary shifts.

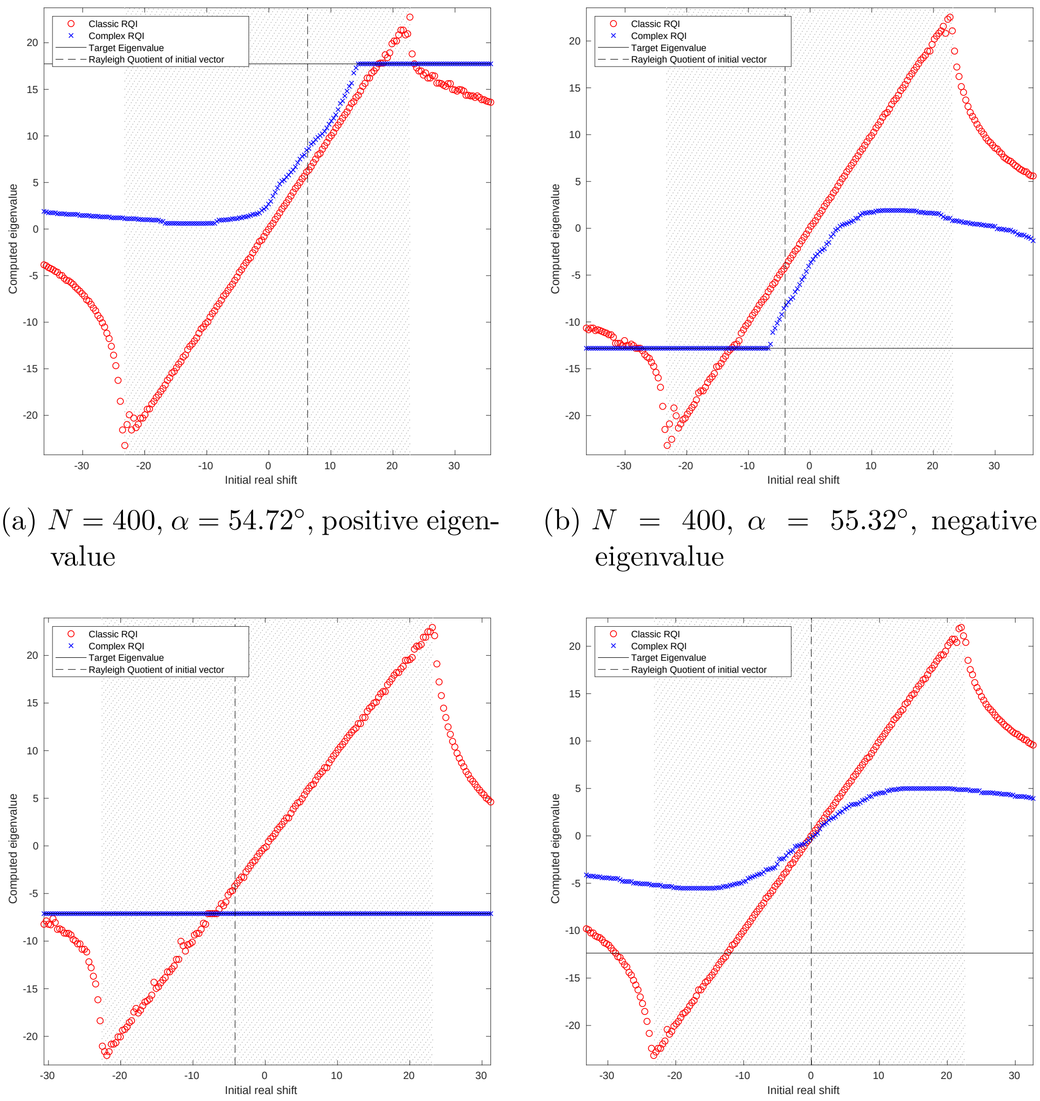
Running the same experiment but increasing the component of the initial vector in the direction of the target eigenvector led to a decrease of the number of additional steps required by CRQI; this observation is analysed in more detail again later. For eigenvectors that were very close to the target sometimes both classic RQI and complex RQI sometimes even finished within the same number of iterations. Still, even in these cases, classic RQI often failed to converge to the right eigenpair.

In the next examples we will examine how the initial vector and initial (real) shift affect the results of RQI and CRQI. We have already discussed the sometimes erratic behaviour of classic RQI. As we will shortly see, a main problem of RQI is that convergence often hardly depends on the initial vector but rather on the initial shift. This is why especially for problems with closely spaced eigenvalues RQI fails to compute the right eigenpair. CRQI seems to be more robust to changes in the initial shift.

We start with an example where the algorithm was run many times with a fixed initial vector but varying initial shifts. To obtain the shifts we computed the spectrum of the matrix using built-in functions of Matlab and extracted 100 evenly spaced values form the interval $[\lambda_{\min} - 10, \lambda_{\max} + 10]$, where $\lambda_{\min}$

and $\lambda_{\max}$ are the smallest and largest eigenvalue of $\boldsymbol{A}$, respectively. The results are plotted in Figure 3.2. The initial vector was set to a weighted combination of the exact eigenvectors as described above. The component in the target direction was small in the first two examples (Figure 3.2(a) and 3.2(b)), big in the third example (Figure 3.2(c)) and not larger than the remaining components in the last example (i. e., none of the weights was set to a larger value than any of the others, Figure 3.2(d)). In all of the remaining figures N denotes the matrix order and α denotes the angle between the initial vector and the target eigenvector.

We observe that RQI seems to depend heavily on the choice of the initial shift, especially when the shift lies between the upper and lower bound of the spectrum (indicated by the dotted area in the plots). This does not change even for initial vectors that are very close approximations of an eigenvector. In contrast, it appears that CRQI does not depend so much on the shift but rather on the initial vector. In the first two examples, it seems as if in some cases the result of CRQI depends on the sign of the target eigenvalue. If the eigenvalue is negative, shifts below this eigenvalue produced the correct result whereas shifts above the eigenvalue did not and analogously for positive eigenvalues. Further investigation revealed that this has actually nothing to do with the target eigenvalue being negative or positive but rather its location in the spectrum (see also the results using the other test matrices given in Figures). If it is below the centre of the spectrum the behaviour is as in Figure 3.2(a) and for eigenvalues above the centre the results are similar to those in Figure 3.2(b). This observation could possibly be used if the initial vector is not that good of an approximation but some knowledge of the spectrum is available so that the initial shifts could be chosen accordingly. If one does not have any a priori knowledge about the distribution of the eigenvalues it seems best to use the algorithm as defined above, i. e., to use the Rayleigh Quotient as the initial shift. Even in cases where the outcome of the method varied with changing initial shift, using the Rayleigh Quotient



(a) $N = 400$, $\alpha = 54.72°$, positive eigenvalue

(b) $N = 400$, $\alpha = 55.32°$, negative eigenvalue

(c) $N = 400$, $\alpha = 38.29°$

(d) $N = 400$, $\alpha = 86.59°$

Figure 3.2: Plot of initial real shift against the computed eigenvalue using classic RQI and CRQI for a random matrix. The dotted area encloses the initial shifts that lie in the spectrum of $\boldsymbol{A}$. The dashed line is located at the Rayleigh quotient of the initial vector.

often produced the correct result, see for instance the first two examples of Figure B.8. Thus, for the remainder of the thesis, CRQI is run with the initial real shift set to the Rayleigh Quotient of the initial vector.

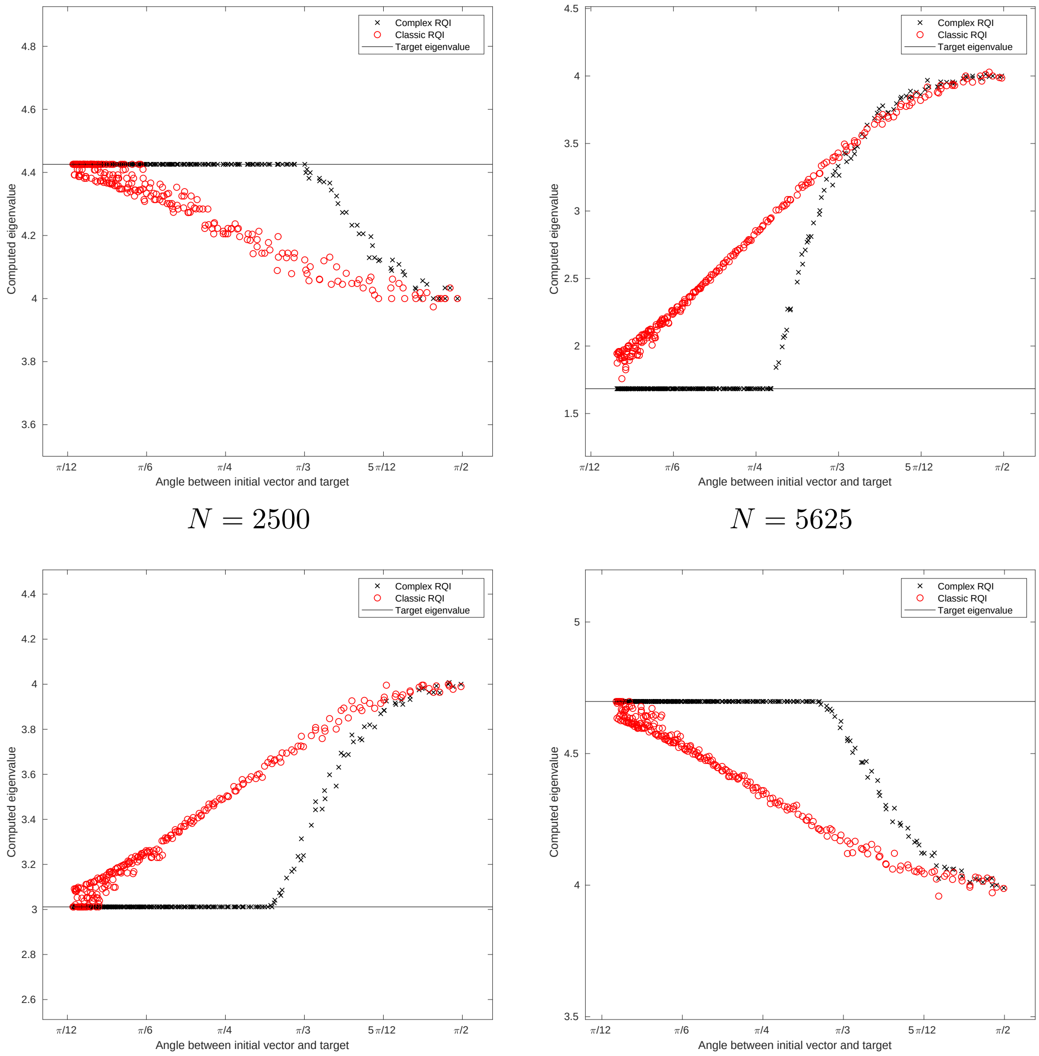
We have introduced CRQI as an improvement of classic RQI in the sense that good approximations of an eigenvector should lead to convergence to this very eigenvector. In the following examples we examine how good of an approximation the initial vector needs to be. We do this by studying the relationship between the angle that the initial vector makes with the target eigenvector and the outcome of the method. To that end, artificial approximations of the wanted eigenvector are created as explained above where the contribution in the target direction is successively increased throughout the experiment. We will see that

(a) classic RQI indeed suffers from the problem that good approximations of an eigenvector do not necessarily lead to convergence to this very eigenvector and that

(b) complex RQI seems to take much more advantage of the information in the initial vector.

In Figure 3.3 we plotted the angle between the initial vector and the wanted eigenvector and the computed eigenvalue for classic RQI and CRQI. This particular example shows the results for the Laplace matrices of different sizes; the results for the remaining test problems are given in Figures B.9 to B.11 in the Appendix.

We see that complex RQI seems to converge to the right eigenvector once the angle between the initial vector and the target is sufficiently small; classic RQI, on the other hand, still often converges to the wrong eigenpair for very small angles. The figure suggests that "sufficiently small" means approximately below $\pi/4$; sometimes higher angles also produce the correct result but taking into account all the expirments we found that it seems almost impossible to make CRQI fail for angles below $\pi/4$.



$N = 2500$

$N = 5625$

$N = 6400$

$N = 10000$

Figure 3.3: Plot of the computed eigenvalue against the angle between the initial vector and the target eigenvector for the Laplace matrix. While classic RQI even fails for very small angles, CRQI produces the correct result as soon as the angle is sufficiently small, which in these examples means below approximately $\pi/4$.

3.3 Implemenation and Practical Considerations

4 Concluding Remarks

Appendices

A Proofs

Proof of Lemma 3.1. First consider $j \neq k$. Using the orthogonality of the eigenvectors we have $\mathbf{v}_k^\mathsf{T}\mathbf{v}_j = 0$ and thus

$$\begin{aligned}(\boldsymbol{A} - i\gamma(\boldsymbol{I} - \mathbf{v}_k\mathbf{v}_k^\mathsf{T}))\mathbf{v}_j &= \boldsymbol{A}\mathbf{v}_j - i\gamma\mathbf{v}_j + i\gamma\mathbf{v}_k\mathbf{v}_k^\mathsf{T}\mathbf{v}_j \\ &= \lambda_j\mathbf{v}_j - i\gamma\mathbf{v}_j \\ &= (\lambda_j - i\gamma)\mathbf{v}_j\,.\end{aligned}$$

For $j = k$ we have $\mathbf{v}_k^\mathsf{T}\mathbf{v}_j = \mathbf{v}_k^\mathsf{T}\mathbf{v}_k = 1$ and thus

$$(\boldsymbol{A} - i\gamma(\boldsymbol{I} - \mathbf{v}_k\mathbf{v}_k^\mathsf{T}))\mathbf{v}_k = \lambda_k\mathbf{v}_k - i\gamma\mathbf{v}_k + i\gamma\mathbf{v}_k = \lambda_k\mathbf{v}_k\,.$$

□

Proof of Lemma 3.2. Since

$$\left\|\tilde{\boldsymbol{A}}_{(1)}\right\| = \left\|i\gamma\left(\mathbf{u}\mathbf{u}^\mathsf{T} - \mathbf{v}_k\mathbf{v}_k^\mathsf{T}\right)\right\| = \gamma\|\underbrace{\mathbf{u}\mathbf{u}^\mathsf{T} - \mathbf{v}_k\mathbf{v}_k^\mathsf{T}}_{=:\boldsymbol{E}}\|$$

it suffices to compute the norm of $\boldsymbol{E}$. Additionally, $\boldsymbol{E}$ is symmetric (since it is a sum of two symmetric matrices) and thus its spectral norm coincides with its spectral radius. The matrices $\mathbf{u}\mathbf{u}^\mathsf{T}$ and $\mathbf{v}_k\mathbf{v}_k^\mathsf{T}$ have both rank one and therefore $\boldsymbol{E}$ is at most of rank two. This implies that it has at most two non-zero eigenvalues, denoted by μ_1 and μ_2, and we can therefore compute both of them and take the one that is larger in modulus.

To compute the eigenvalues of $\boldsymbol{E}$ we use the following two well-known facts from linear algebra. Here, $\mathrm{tr}(\boldsymbol{E})$ denotes the trace of $\boldsymbol{E}$ defined as the sum of the diagonal entries of the matrix.

1. The trace is the sum of the eigenvalues of $\boldsymbol{E}$, i. e.,

eigenvalues * algebraic multiplicity

$$\operatorname{tr}(\boldsymbol{E}) = \mu_1 + \mu_2\,. \tag{A.1}$$

2. The eigenvalues of the square of $\boldsymbol{E}$ are μ_1^2 and μ_2^2 and thus, using the previous fact,
$$\operatorname{tr}(\boldsymbol{E}^2) = \mu_1^2 + \mu_2^2\,. \tag{A.2}$$

The diagonal entries of the matrices $\mathbf{u}\mathbf{u}^\mathsf{T}$ and $\mathbf{v}_k\mathbf{v}_k^\mathsf{T}$ consist of the squares of their respective entries and thus, using the additivity of the trace and the fact that both vectors are assumed to be normalised, we get

$$\operatorname{tr}(\boldsymbol{E}) = \|\mathbf{u}\| - \|\mathbf{v}_k\| = 0$$

which in conjunction with (A.1) implies $\mu_1 = -\mu_2$. We expand $\boldsymbol{E}^2$ and obtain

$$\begin{aligned}\boldsymbol{E}^2 &= (\mathbf{u}\mathbf{u}^\mathsf{T} - \mathbf{v}_k\mathbf{v}_k^\mathsf{T})(\mathbf{u}\mathbf{u}^\mathsf{T} - \mathbf{v}_k\mathbf{v}_k^\mathsf{T})\\ &= \mathbf{u}\mathbf{u}^\mathsf{T}\mathbf{u}\mathbf{u}^\mathsf{T} - \mathbf{v}_k\mathbf{v}_k^\mathsf{T}\mathbf{u}\mathbf{u}^\mathsf{T} - \mathbf{u}\mathbf{u}^\mathsf{T}\mathbf{v_k}\mathbf{v_k}^\mathsf{T} + \mathbf{v}_k\mathbf{v}_k^\mathsf{T}\mathbf{v}_k\mathbf{v}_k^\mathsf{T}\\ &= \mathbf{u}\mathbf{u}^\mathsf{T} + \mathbf{v}_k\mathbf{v}_k^\mathsf{T} - \langle\mathbf{u}, \mathbf{v}_k\rangle\left(\mathbf{u}\mathbf{v}_k^\mathsf{T} + \mathbf{v}_k\mathbf{u}^\mathsf{T}\right)\,.\end{aligned}$$

Therefore,

$$\begin{aligned}\operatorname{tr}(\boldsymbol{E}^2) &= \|\mathbf{u}\| + \|\mathbf{v}_k\| - \langle\mathbf{u}, \mathbf{v}_k\rangle(\langle\mathbf{u}, \mathbf{v}_k\rangle + \langle\mathbf{u}, \mathbf{v}_k\rangle)\\ &= 2 - 2\langle\mathbf{u}, \mathbf{v}_k\rangle^2\,.\end{aligned}$$

However, we also have $\operatorname{tr}(\boldsymbol{E}^2) = 2\mu_1^2$ due to (A.2) and so

$$\mu_1 = \sqrt{1 - \langle\mathbf{u}, \mathbf{v}_k\rangle^2}$$

from which the result follows. □

Proof of Proposition 3.3. For this proof we treat $\tilde{\boldsymbol{A}}_{(1)}$ as a perturbation of $\tilde{\boldsymbol{A}}_{(0)}$ and a result from [**ref**] which states that the eigenvalues of $\tilde{\boldsymbol{A}} = \tilde{\boldsymbol{A}}_{(0)} +$

Find Reference for this fact

$\tilde{\boldsymbol{A}}_{(1)}$ satisfy

$$\lambda_j(\tilde{\boldsymbol{A}}) = \lambda_j(\tilde{\boldsymbol{A}}_{(0)}) + \mathbf{w}_j^* \tilde{\boldsymbol{A}}_{(0)} \mathbf{w}_j + \mathcal{O}\left(\left\|\tilde{\boldsymbol{A}}_{(1)}\right\|^2\right) , \tag{A.3}$$

where $\mathbf{w}_j$ is an eigenvector belonging to $\lambda_j(\tilde{\boldsymbol{A}}_{(0)})$. We expand the middle part and obtain

$$\begin{aligned}
\mathbf{w}_j^* \tilde{\boldsymbol{A}}_{(0)} \mathbf{w}_j &= i\gamma \mathbf{w}_j^* \left(\mathbf{u}\mathbf{u}^* - \mathbf{v}_k \mathbf{v}_k^*\right) \mathbf{w}_j \\
&= i\gamma \left(\mathbf{w}_j^* \mathbf{u}\mathbf{u}^* \mathbf{w}_j - \mathbf{w}_j^* \mathbf{v}_k \mathbf{v}_k^* \mathbf{w}_j\right) \\
&= i\gamma \left(\langle \mathbf{w}_j, \mathbf{u}\rangle^2 - \langle \mathbf{w}_j, \mathbf{v}_k\rangle^2\right) \\
&= \begin{cases} i\gamma \langle \mathbf{v}_j, \mathbf{u}\rangle^2 & \text{if } j \neq k\,, \\ i\gamma \left(\langle \mathbf{v}_j, \mathbf{u}\rangle^2 - 1\right) & \text{if } j = k\,, \end{cases}
\end{aligned}$$

where we used Lemma 3.1 in the last step, which states that in fact $\mathbf{w}_j = \mathbf{v}_j$ with $\mathbf{v}_j$ being the eigenvectors of $\boldsymbol{A}$. Adding this to the first part of (A.3) yields

$$\begin{aligned}
\lambda_j(\tilde{\boldsymbol{A}}_{(0)}) + \mathbf{w}_j^* \tilde{\boldsymbol{A}}_{(0)} \mathbf{w}_j &= \lambda_j(\boldsymbol{A}) - i\gamma + i\gamma \langle \mathbf{v}_j, \mathbf{u}\rangle^2 \\
&= \lambda_j(\boldsymbol{A}) + i\gamma \left(\langle \mathbf{v}_j, \mathbf{u}\rangle^2 - 1\right)
\end{aligned}$$

for $j \neq k$ and

$$\lambda_j(\tilde{\boldsymbol{A}}_{(0)}) + \mathbf{w}_j^* \tilde{\boldsymbol{A}}_{(0)} \mathbf{w}_j = \lambda_k(\boldsymbol{A}) + i\gamma \left(\langle \mathbf{v}_k, \mathbf{u}\rangle^2 - 1\right)$$

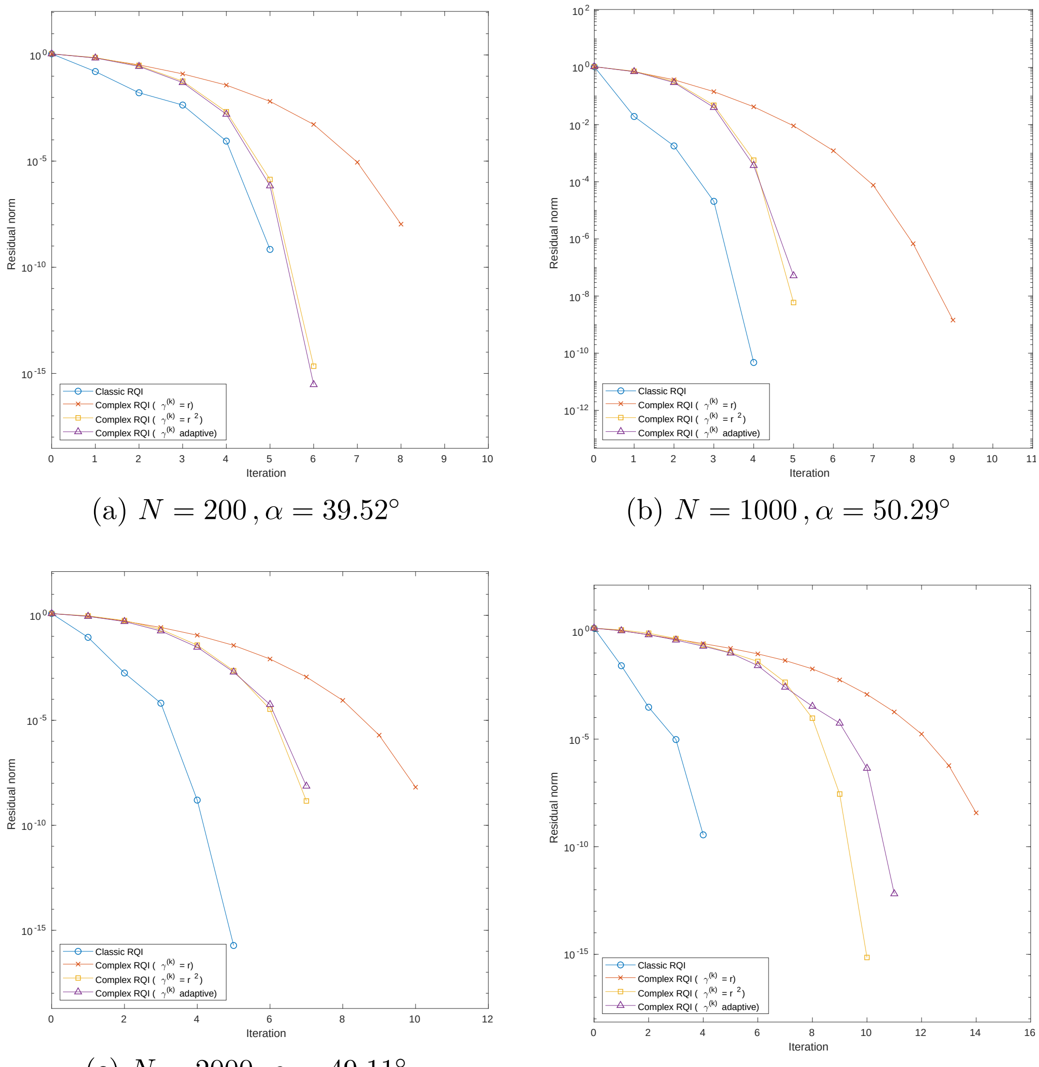
for $j = k$ and thus

$$\lambda_j(\tilde{\boldsymbol{A}}) = \lambda_j(\boldsymbol{A}) + i\gamma \left(\langle \mathbf{v}_j, \mathbf{u}\rangle^2 - 1\right) + \mathcal{O}\left(\left\|\tilde{\boldsymbol{A}}_{(1)}\right\|^2\right) ,$$

for all $j = 1, \ldots, n$, which concludes the proof. □

B Additional Plots

In this chapter we include some additional plots for the examples of Section 3.2. We fix the following notation: N denotes the order of the test matrix that was used in the respective example, i. e., the size of the matrix is $N \times N$. The (acute) angle between the target eigenvector and the initial vector that was used in the algorithm is denoted by α (the angle is always given in degrees).



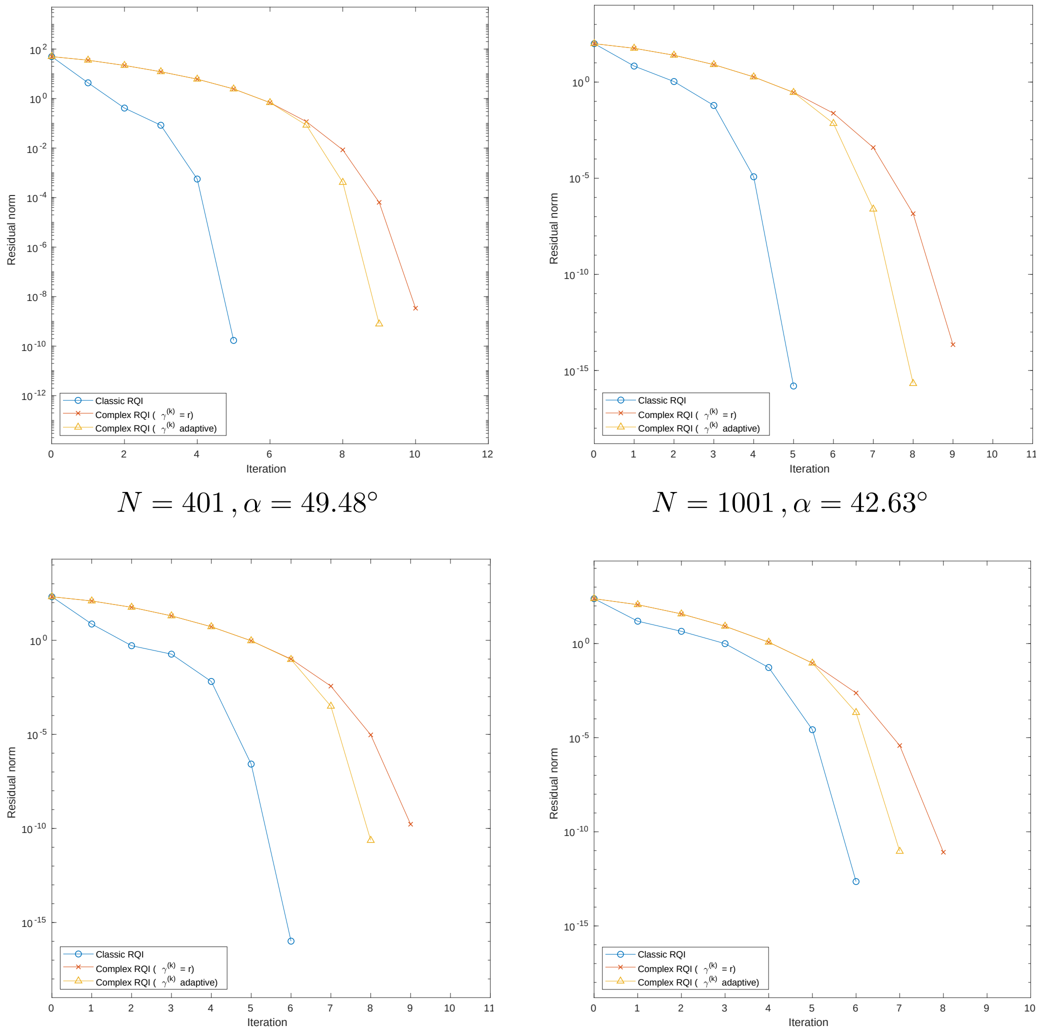
(a) $N = 200\,, \alpha = 39.52°$

(b) $N = 1000\,, \alpha = 50.29°$

(c) $N = 2000\,, \alpha = 49.11°$

$N = 4000\,, \alpha = 46.29°$

Figure B.1: Plot of the residual for different choices of imaginary shifts for the $[1, 2, 1]$ matrix



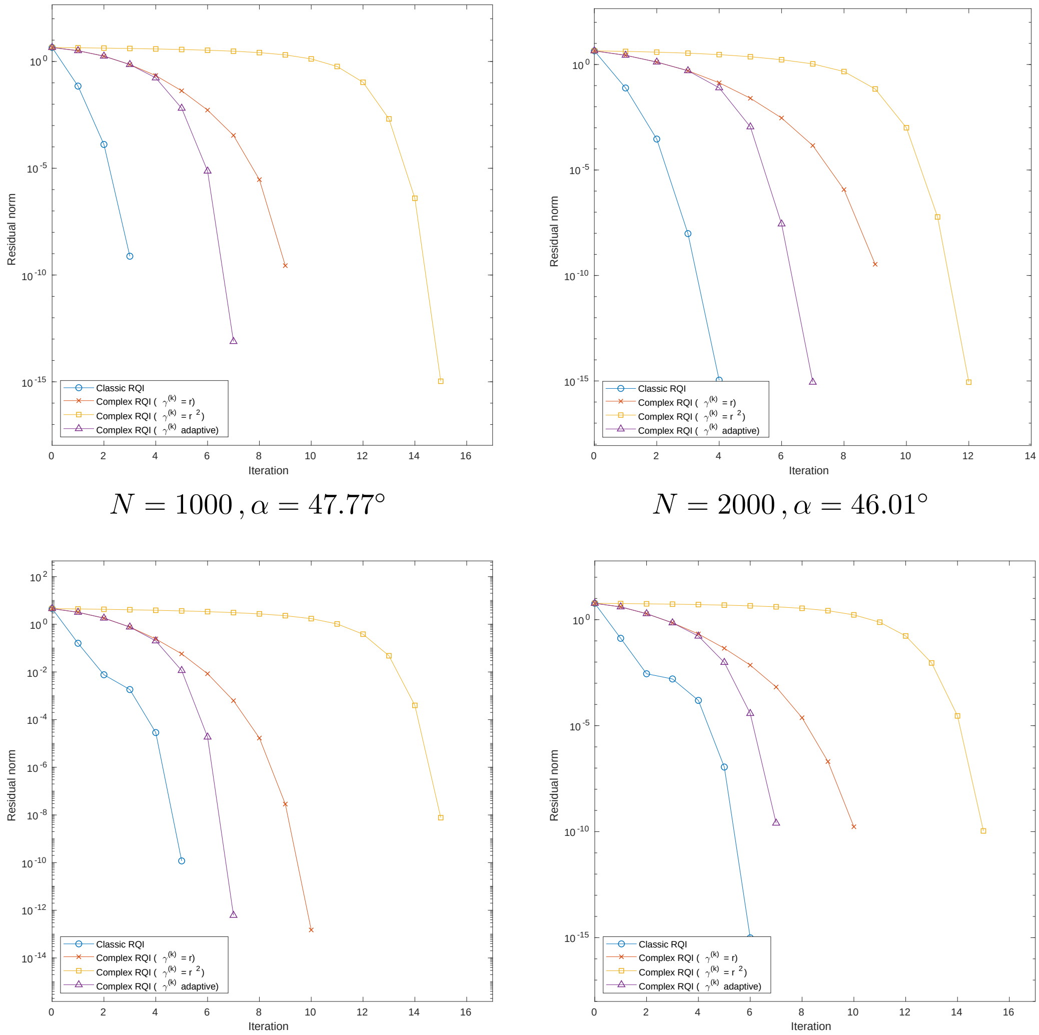
$N = 401\,,\alpha = 49.48°$

$N = 1001\,,\alpha = 42.63°$

$N = 2001\,,\alpha = 45.29°$

$N = 3001\,,\alpha = 33.89°$

Figure B.2: Plot of the residual for different choices of imaginary shifts for the Wilkinson matrix $\boldsymbol{W}^{+}$. The results using the squares residual norm as the imaginary shift are not included since in all cases the number of iterations was too high.



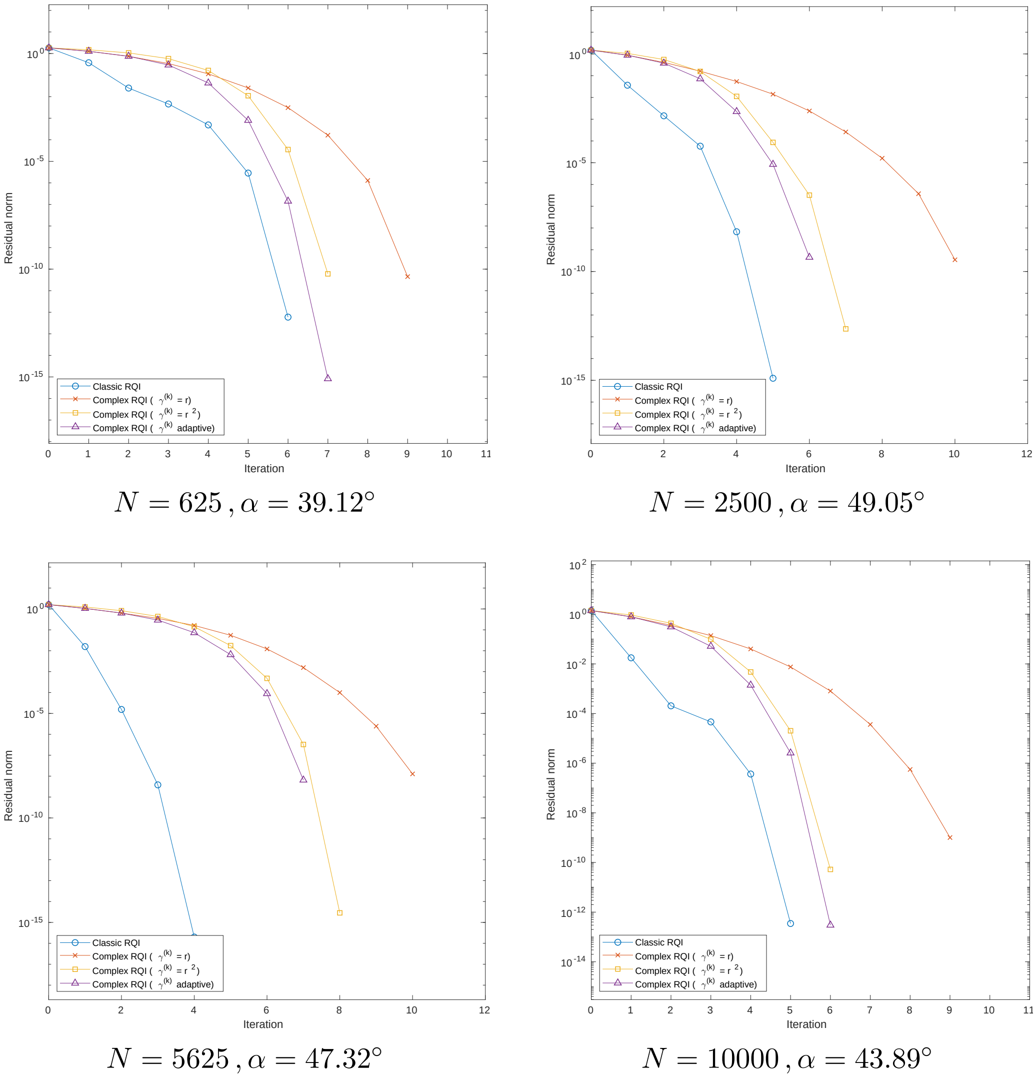
$N = 1000\,, \alpha = 47.77°$

$N = 2000\,, \alpha = 46.01°$

$N = 4000\,, \alpha = 50.79°$

$N = 5000\,, \alpha = 36.70°$

Figure B.3: Plot of the residual for different choices of imaginary shifts for the Martin-Wilkinson matrix $\boldsymbol{MW}$.



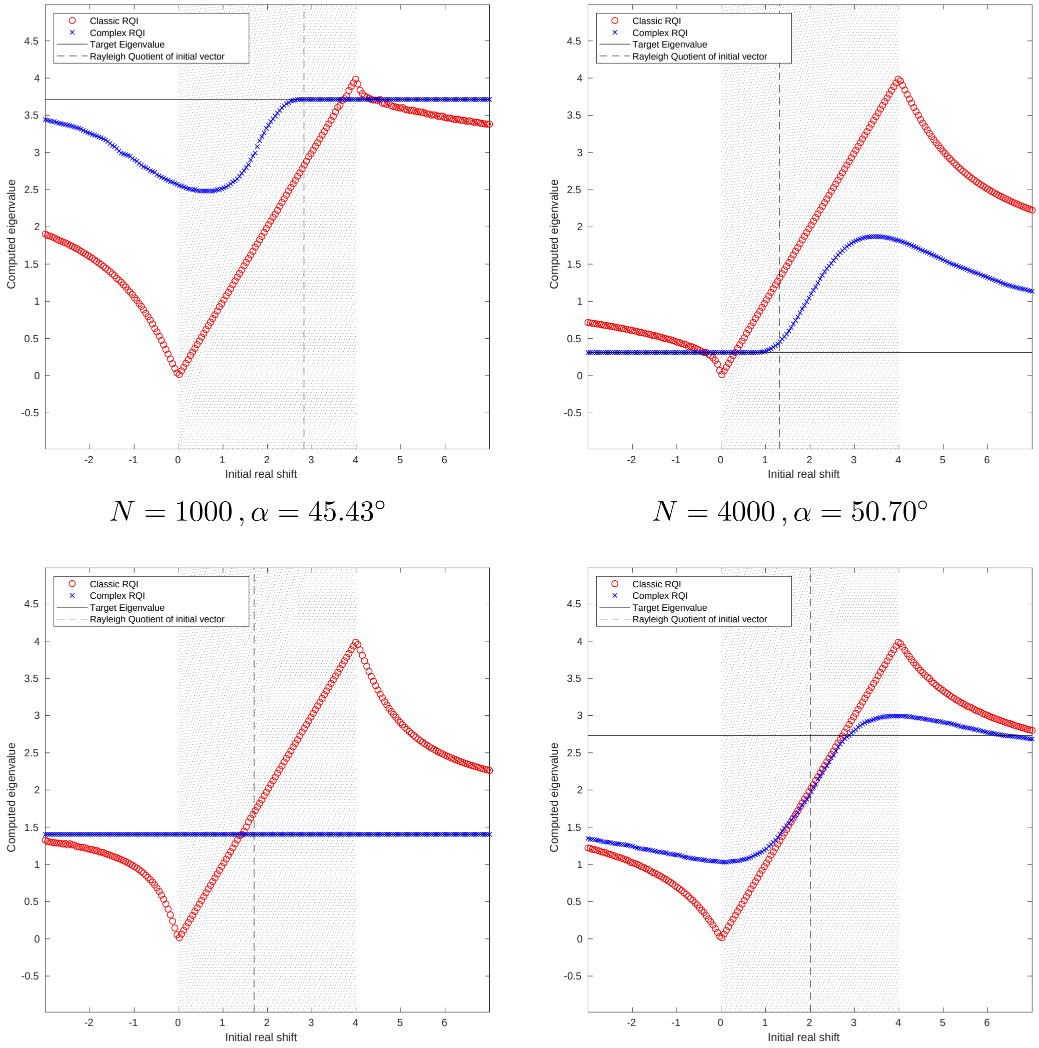
$N = 625\,, \alpha = 39.12°$

$N = 2500\,, \alpha = 49.05°$

$N = 5625\,, \alpha = 47.32°$

$N = 10000\,, \alpha = 43.89°$

Figure B.4: Plot of the residual for different choices of imaginary shifts for the Laplace matrix.



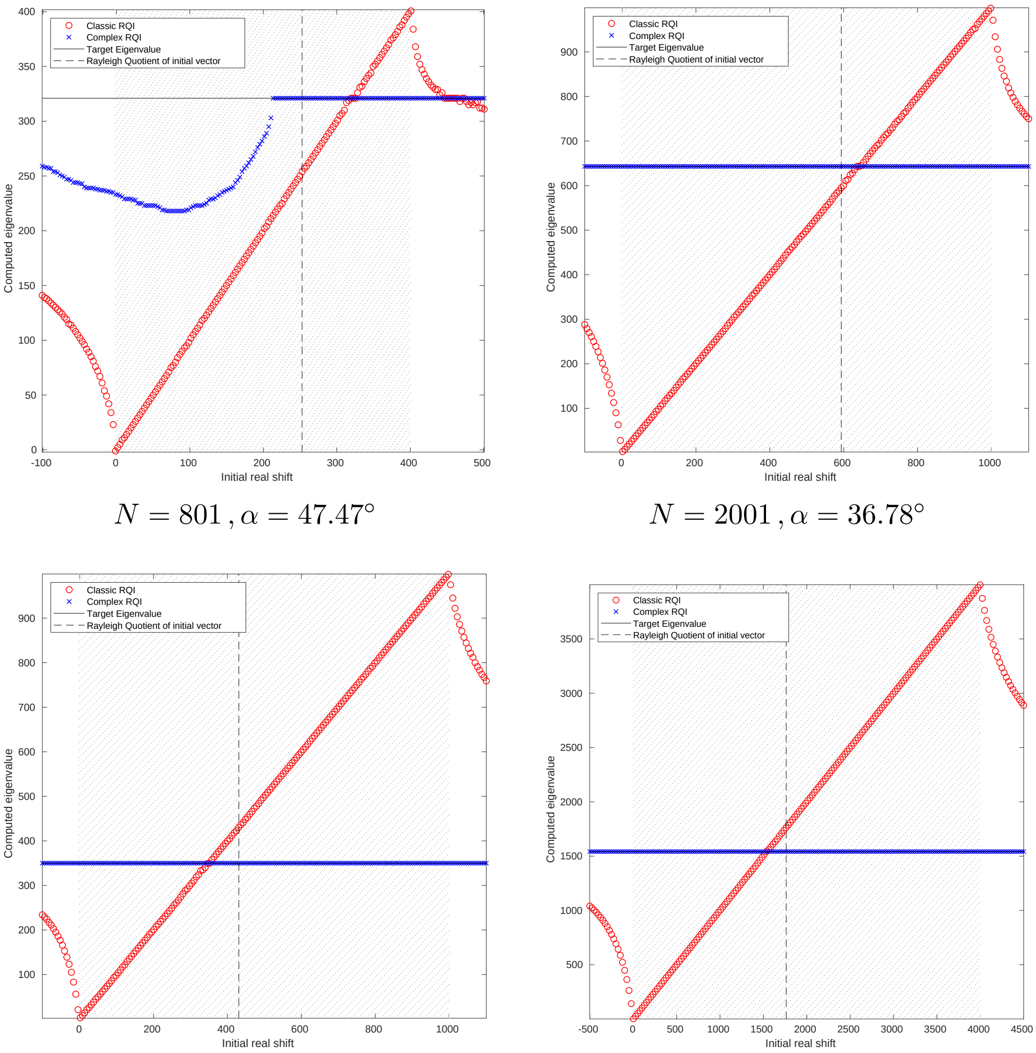
$N = 1000\,, \alpha = 45.43°$

$N = 4000\,, \alpha = 50.70°$

$N = 4000\,, \alpha = 46.23°$

$N = 1000\,, \alpha = 86.83°$

Figure B.5: Plot of the computed eigenvalue against the initial real shift for the $[1, 2, 1]$ matrix.



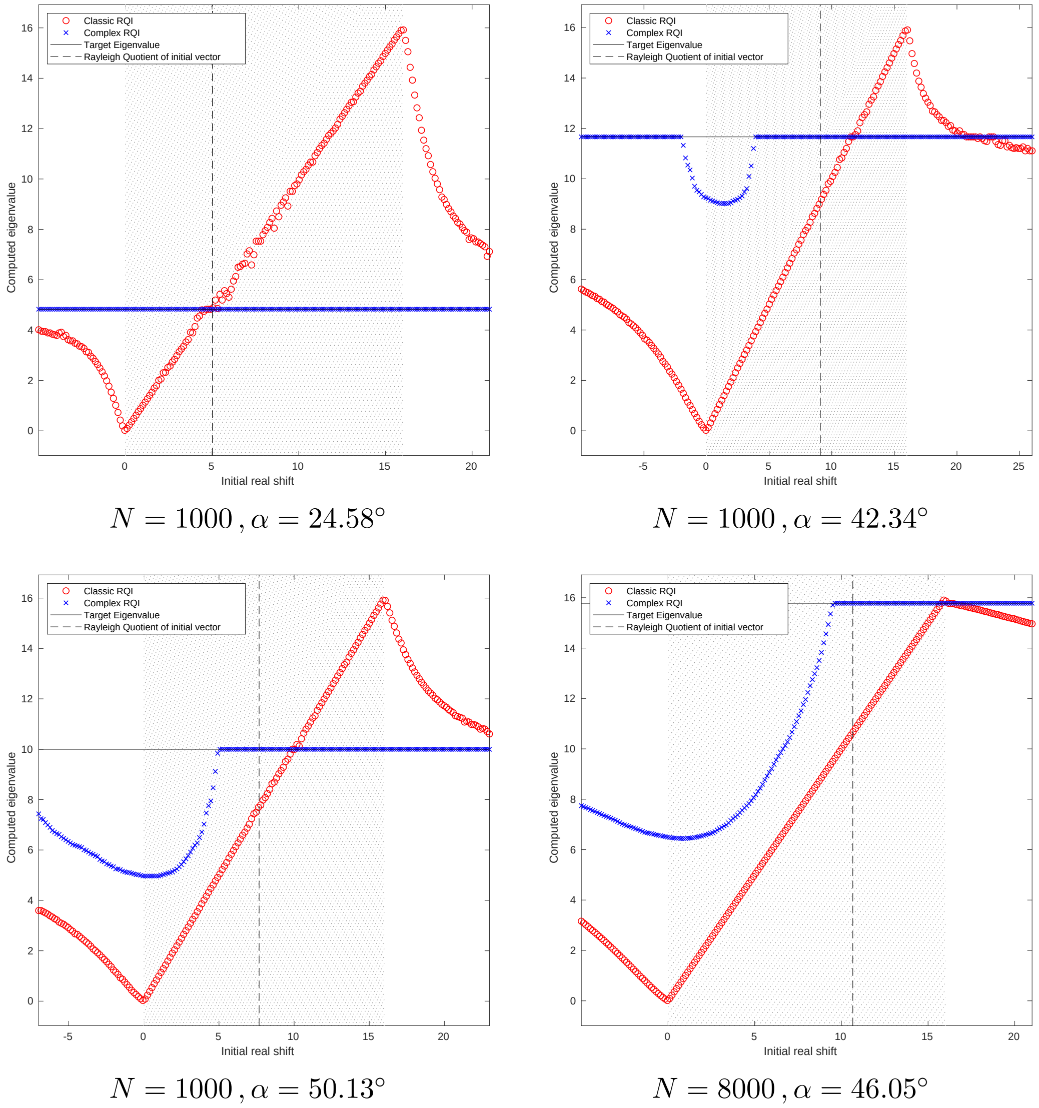
$N = 801\,, \alpha = 47.47°$

$N = 2001\,, \alpha = 36.78°$

$N = 2001\,, \alpha = 45.82°$

$N = 8001\,, \alpha = 43.69°$

Figure B.6: Plot of the computed eigenvalue against the initial real shift for the Wilkinson matrix $\boldsymbol{W}^+$.



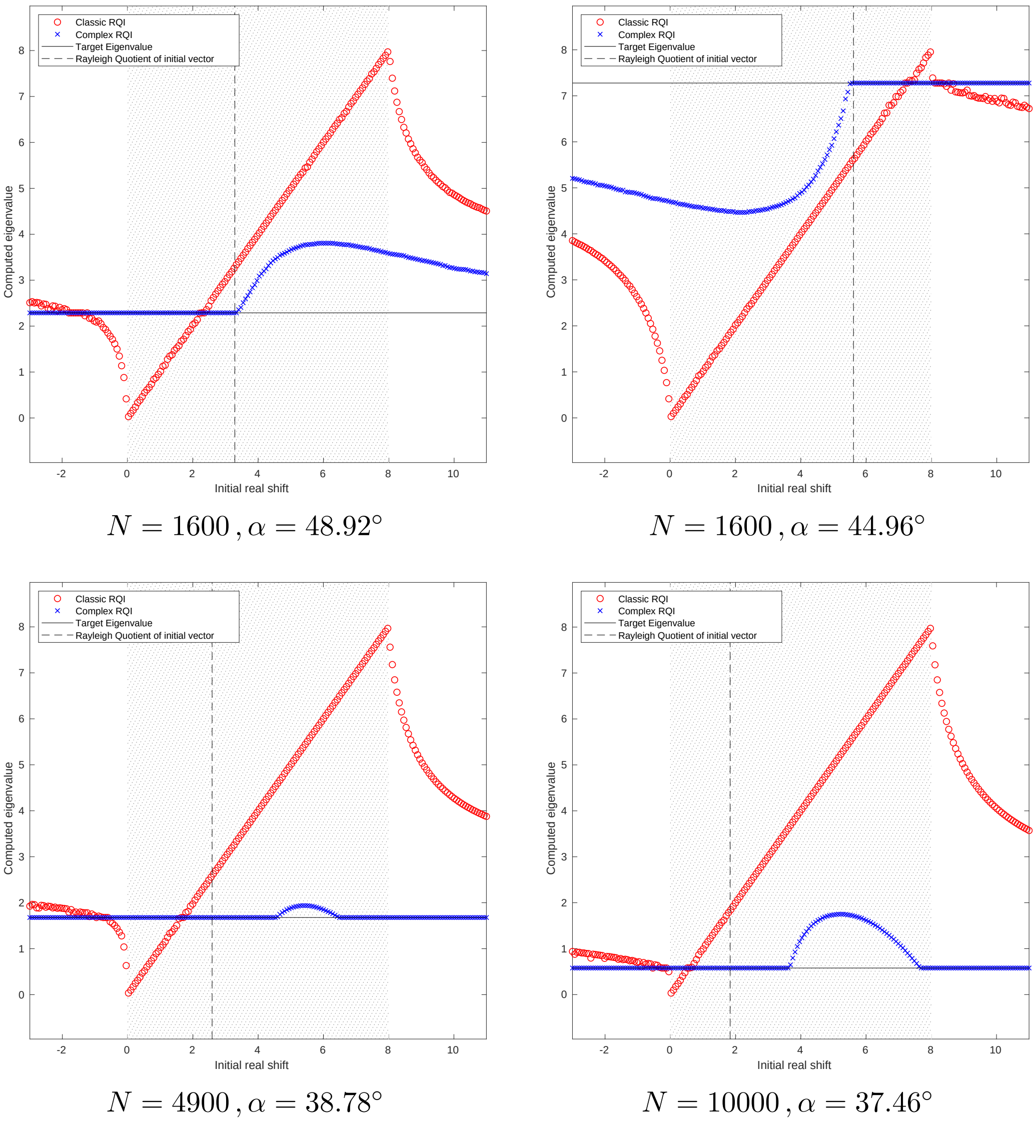
$N = 1000\,, \alpha = 24.58°$

$N = 1000\,, \alpha = 42.34°$

$N = 1000\,, \alpha = 50.13°$

$N = 8000\,, \alpha = 46.05°$

Figure B.7: Plot of the computed eigenvalue against the initial real shift for the Martin-Wilkinson matrix $\boldsymbol{MW}$.



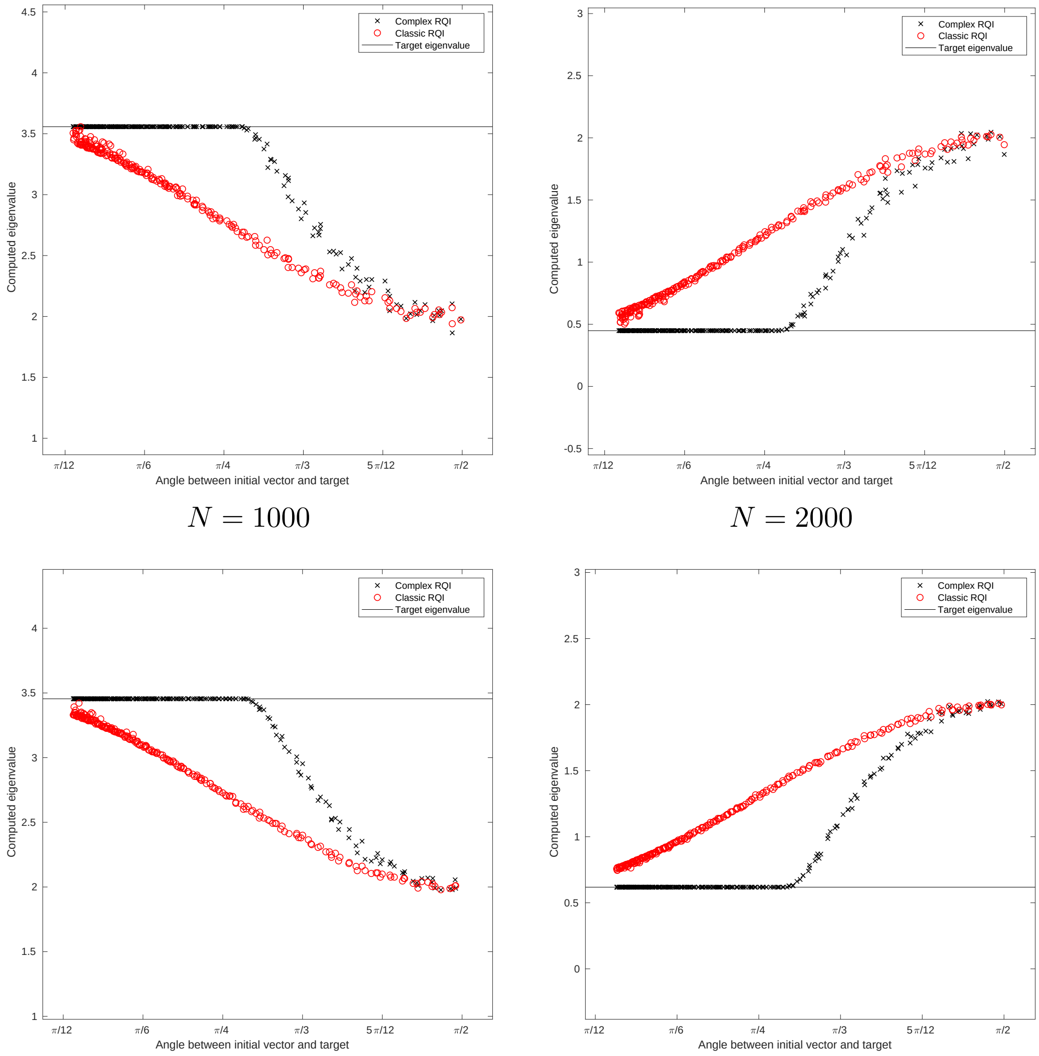
$N = 1600\,, \alpha = 48.92°$

$N = 1600\,, \alpha = 44.96°$

$N = 4900\,, \alpha = 38.78°$

$N = 10000\,, \alpha = 37.46°$

Figure B.8: Plot of the computed eigenvalue against the initial real shift for the Laplace matrix.



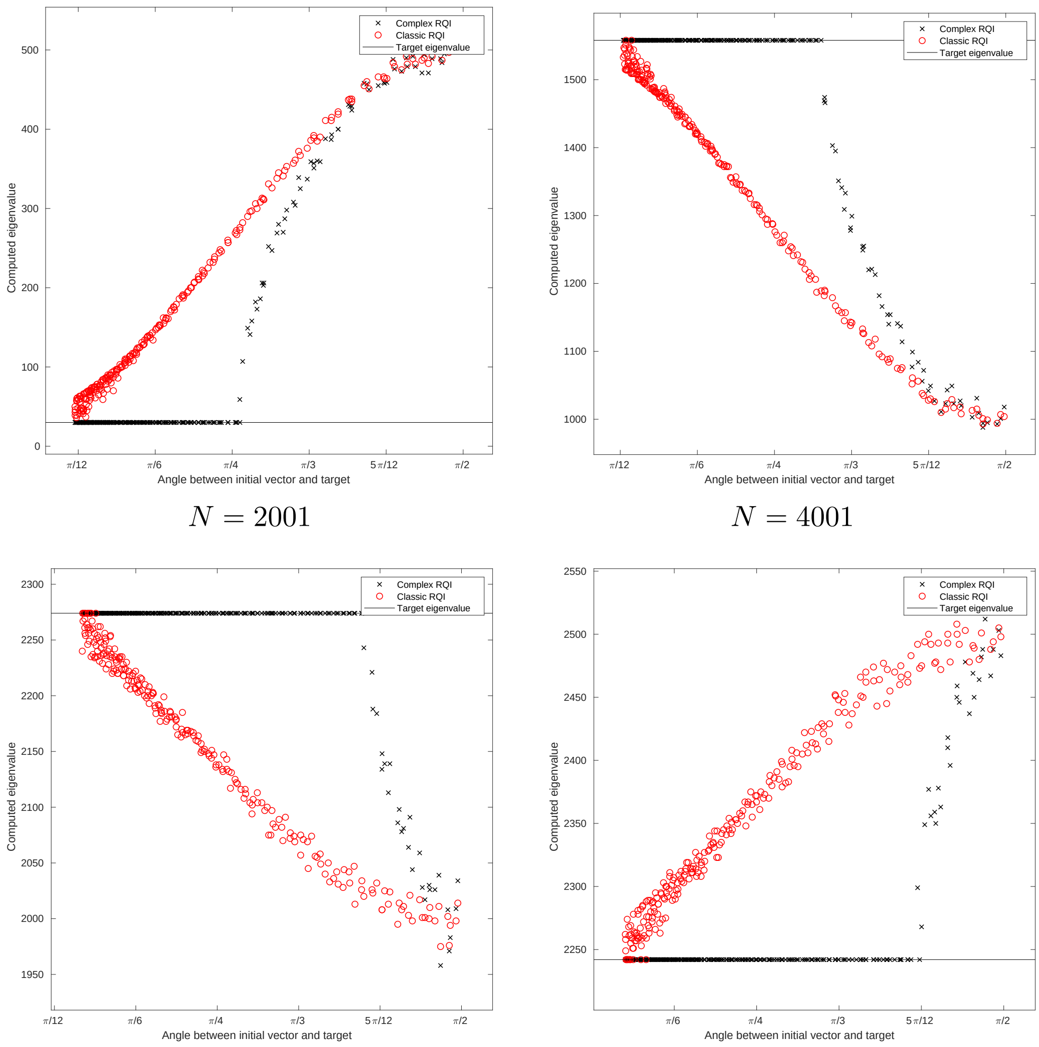
$N = 1000$

$N = 2000$

$N = 4000$

$N = 8000$

Figure B.9: Plot of the computed eigenvalue against the angle between the initial vector and the target eigenvector for the matrix $[1, 2, 1]$.



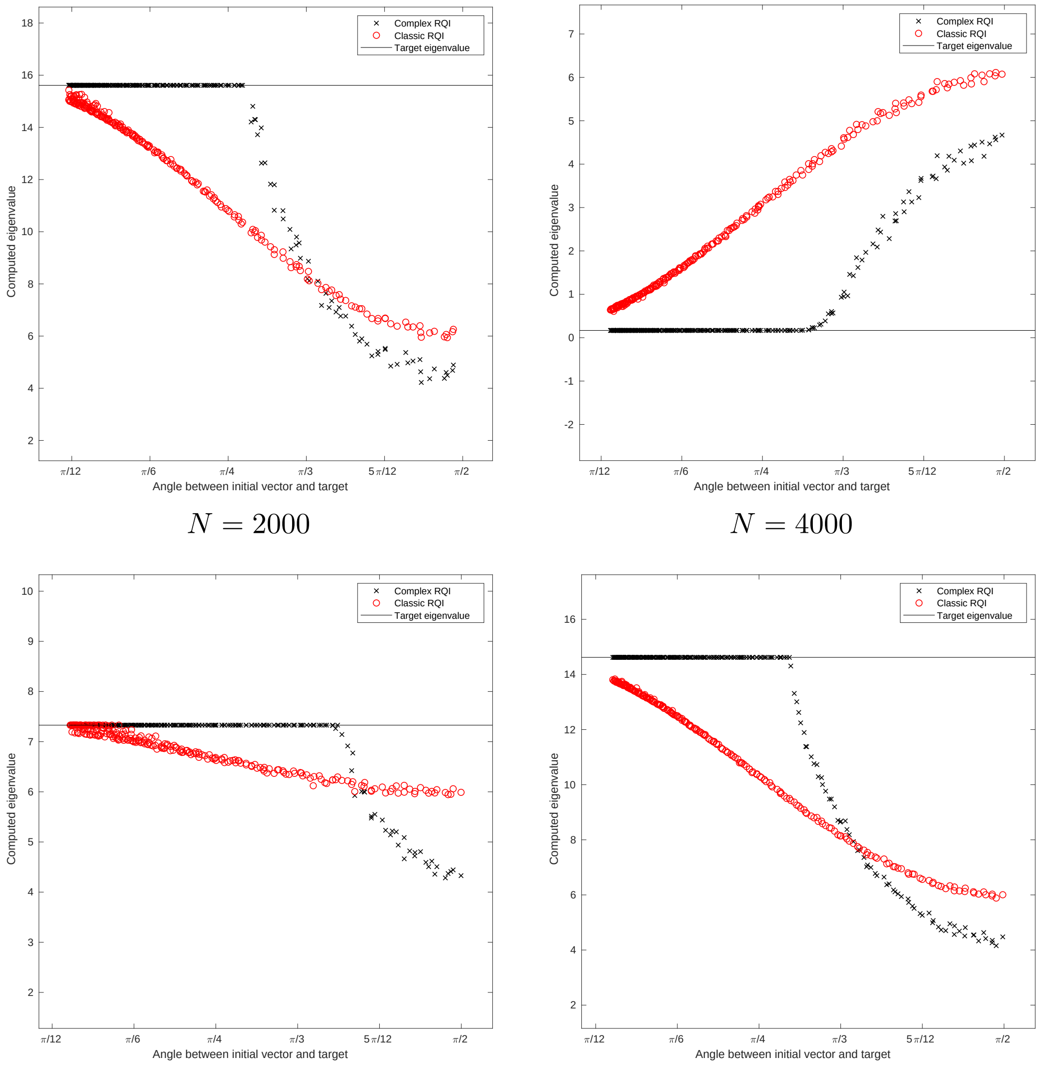
$N = 2001$

$N = 4001$

$N = 8001$

$N = 10001$

Figure B.10: Plot of the computed eigenvalue against the angle between the initial vector and the target eigenvector for the Wilkinson matrix $\boldsymbol{W}^+$.



$N = 2000$

$N = 4000$

$N = 8000$

$N = 10000$

Figure B.11: Plot of the computed eigenvalue against the angle between the initial vector and the target eigenvector for the Martin-Wilkinson matrix $\boldsymbol{MW}$.

Bibliography

[1] Steve Batterson and John Smillie. “The Dynamics of Rayleigh Quotient Iteration”. In: *SIAM Journal on Numerical Analysis* 25.3 (1989), pp. 624–636 (cit. on pp. 30, 38).

[2] Christopher Beattie and David W. Fox. “Localization Criteria and Containment for Rayleigh Quotient Iteration”. In: *SIAM Journal on Matrix Analysis and Applications* 10.1 (1989), pp. 80–93 (cit. on p. 29).

[3] Steffen Börm. *Numerik von Eigenwertaufgaben.* Lecture notes. 2018 (Last accessed: 14 March 2020). URL: `https://www.informatik.uni-kiel.de/~sb/data/Eigenwerte.pdf` (cit. on pp. 22, 35).

[4] Stephen H. Crandall and Richard Vynne Southwell. “Iterative procedures related to relaxation methods for eigenvalue problems”. In: *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* 207.1090 (1951), pp. 416–423 (cit. on p. 25).

[5] Achiya Dax. “The Orthogonal Rayleigh Quotient Iteration (ORQI) method”. In: *Linear Algebra and its Applications* 358.1 (2003), pp. 23–43 (cit. on p. 46).

[6] James W. Demmel. *Applied Numerical Linear Algebra.* Philadelphia: Society for Industrial and Applied Mathematics, 1997 (cit. on pp. 31, 48).

[7] Melina A. Freitag and Alastair Spence. “Rayleigh quotient iteration and simplified Jacobi–Davidson method with preconditioned iterative solves”. In: *Linear Algebra and its Applications* 428 (2008), pp. 2049–2060 (cit. on p. 31).

[8] Stephen H. Friedberg, Arnold J. Insel, and Lawrence E. Spence. *Linear Algebra: Pearson New International Edition.* Fourth Edition. Pearson Education, 2014 (cit. on p. 20).

[9] I. Galligani. *A comparison of methods for computing the eigenvalues and eigenvectors of a matrix.* Tech. rep. European Atomic Energy Community – Euratom, 1968 (cit. on p. 48).

[10] Gene H. Golub and Henk A. Van der Vorst. "Eigenvalue computation in the 20th century". In: *Journal of Computational and Applied Mathematics* 123.1-2 (2000), pp. 35–65 (cit. on pp. 8, 17).

[11] Gene H. Golub and Charles F. Van Loan. *Matrix Computations.* Third Edition. Baltimore and London: The John Hopkins University Press, 1996 (cit. on p. 44).

[12] Kenneth M. Hoffman and Ray A. Kunze. *Linear Algebra.* Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1971 (cit. on p. 10).

[13] Ilse C. F. Ipsen. "A History Of Inverse Iteration". In: *Helmut Wielandt, Mathematische Werke, Mathematical Works, Volume 2: Linear Algebra and Analysis.* Ed. by Bertram Huppert and Hans Schneider. Berlin, New York: Walter de Gruyter, 1996, pp. 464–472 (cit. on p. 16).

[14] Carl Gustav Jacob Jacobi. "Über ein leichtes Verfahren die in der Theorie der Säcularstörungen vorkommenden Gleichungen numerisch aufzulösen". In: *Journal für die reine und angewandte Mathematik (Crelles Journal)* 30 (1846), pp. 51–94 (cit. on p. 8).

[15] Walter Kohn. "A Variational Iteration Method for Solving Secular Equations". In: *The Journal of Chemical Physics* 17.7 (1949), p. 670 (cit. on p. 24).

[16] *MATLAB version 9.6.0.1072779 (R2019a).* The Mathworks, Inc. Natick, Massachusetts, 2019 (cit. on p. 46).

[17] Yvan Notay. “Convergence Analysis of Inexact Rayleigh Quotient Iteration”. In: *SIAM Journal on Matrix Analysis and Applications* 24.23 (2003), pp. 627–644 (cit. on p. 30).

[18] Alexander M. Ostrowski. “On the convergence of the Rayleigh quotient iteration for the computation of the characteristic roots and vectors. I”. In: *Archive for Rational Mechanics and Analysis* 1.1 (1957), pp. 233–241 (cit. on pp. 25, 27, 29, 31).

[19] Alexander M. Ostrowski. “On the convergence of the Rayleigh quotient iteration for the computation of the characteristic roots and vectors. II”. In: *Archive for Rational Mechanics and Analysis* 2.1 (1958), pp. 423–428 (cit. on pp. 25, 27).

[20] Alexander M. Ostrowski. “On the convergence of the Rayleigh quotient iteration for the computation of the characteristic roots and vectors. III”. In: *Archive for Rational Mechanics and Analysis* 3.1 (1959), pp. 325–340 (cit. on pp. 25, 27).

[21] Alexander M. Ostrowski. “On the convergence of the Rayleigh quotient iteration for the computation of the characteristic roots and vectors. IV”. In: *Archive for Rational Mechanics and Analysis* 3.1 (1959), pp. 341–347 (cit. on p. 25).

[22] Alexander M. Ostrowski. “On the convergence of the Rayleigh quotient iteration for the computation of the characteristic roots and vectors. V”. In: *Archive for Rational Mechanics and Analysis* 3.1 (1959), pp. 472–481 (cit. on p. 25).

[23] Alexander M. Ostrowski. “On the convergence of the Rayleigh quotient iteration for the computation of the characteristic roots and vectors. VI”. In: *Archive for Rational Mechanics and Analysis* 4.1 (1959), pp. 153–165 (cit. on p. 25).

[24] Ricardo D. Pantazis and Daniel B. Szyld. “Regions of convergence of the Rayleigh quotient iteration method”. In: *Numerical Linear Algebra with Applications* 2.3 (1995), pp. 251–269 (cit. on pp. 28, 30).

[25] Beresford N. Parlett. “The Rayleigh Quotient Iteration and Some Generalizations for Nonnormal Matrices”. In: *Mathematics of Computation* 28.127 (1974), pp. 679–693 (cit. on pp. 27, 38).

[26] Beresford N. Parlett. *The Symmetric Eigenvalue Problem.* Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1998 (cit. on pp. 16, 17, 21, 28, 30, 32, 35, 37, 38).

[27] Beresford N. Parlett and William Kahan. “On the convergence of a practical QR algorithm”. In: *Information Processing, Proceedings of IFIP Congress 1968, Edinburgh, UK, 5-10 August 1968, Volume 1 - Mathematics, Software.* Ed. by A. J. H. Morrel. 1968, pp. 114–118 (cit. on pp. 27, 35).

[28] G. Peters and James H. Wilkinson. “Inverse Iteration, Ill-Conditioned Equations and Newton’s Method”. In: *SIAM Review* 21.3 (1979), pp. 339–360 (cit. on p. 16).

[29] John William Strutt, Baron Rayleigh. *The Theory of Sound.* 2nd ed. Vol. 1. London and New York: MacMillan and Co, 1894 (cit. on p. 24).

[30] Joost Rommes. “Methods for eigenvalue problems with applications in model order reduction”. PhD thesis. Utrecht University, 2007 (cit. on p. 29).

[31] Yousef Saad. *Numerical Methods for Large Eigenvalue Problems.* Society for Industrial and Applied Mathematics, 2011 (cit. on pp. 14, 17, 22, 23).

[32] Valeria Simoncini and Lars Eldén. “Inexact Rayleigh Quotient-Type Methods for Eigenvalue Computations”. In: *BIT Numerical Mathematics* 42.1 (2002), pp. 159–182 (cit. on p. 30).

[33] Daniel B. Szyld. “Criteria for Combining Inverse and Rayleigh Quotient Iteration”. In: *SIAM Journal on Numerical Analysis* 25.6 (1988), pp. 1369–1375 (cit. on pp. 29, 30).

[34] Richard A. Tapia, J. E. Dennis, and Jan P. Schäfermeyer. “Inverse, Shifted Inverse, and Rayleigh Quotient Iteration as Newtons Method”. In: *SIAM Review* 60.1 (2018), pp. 3–55 (cit. on pp. 16, 24, 25).

[35] Lloyd N. Trefethen and David Bau. *Numerical linear algebra.* Other titles in applied mathematics. Society for Industrial and Applied Mathematics, 1997 (cit. on pp. 8, 21, 35).

[36] Helmut Wielandt. *Beiträge zur mathematischen Behandlung komplexer Eigenwertprobleme, Teil V: Bestimmung höherer Eigenwerte durch gebrochene Iteration.* Bericht B 44/J/37. Aerodynamische Versuchsanstalt Göttingen, 1944 (cit. on p. 26).

[37] James H. Wilkinson. *The Algebraic Eigenvalue Problem.* Ed. by E. T. Goodwin and L Fox. Monographs On Numerical Analysis. Oxford: Clarendon Press, 1965 (cit. on pp. 16, 47, 48).