INVERSE, SHIFTED INVERSE, AND RAYLEIGH QUOTIENT ITERATION AS NEWTON'S METHOD[*]

JOHN DENNIS JR.[†] AND RICHARD TAPIA[‡]

Abstract. Two-norm normalized inverse, shifted inverse, and Rayleigh quotient iteration are well-known algorithms for approximating an eigenvector of a symmetric matrix. In this work we establish rigorously that each one of these three algorithms can be viewed as a standard form of Newton's method from the nonlinear programming literature, followed by the normalization. This equivalence adds considerable understanding to the formal structure of inverse, shifted inverse, and Rayleigh quotient iteration and provides an explanation for their good behavior despite the possible need to solve systems with nearly singular coefficient matrices; the algorithms have what can be viewed as removable singularities. A thorough historical development of these eigenvalue algorithms is presented. Utilizing our equivalences we construct traditional Newton's method theory analysis in order to gain understanding as to why, as normalized Newton's method, inverse iteration and shifted inverse iteration are only linearly convergent and not quadratically convergent, and why a new linear system need not be solved at each iteration. We also investigate why Rayleigh quotient iteration is cubically convergent and not just quadratically convergent.

Key words. inverse power method, inverse iteration, shifted inverse iteration, Rayleigh quotient iteration, Newton's method

AMS subject classifications. 65F15, 49M37, 49M15, 65K05

1. Introduction.

The initial objective of this study was to answer the following age-old question: In what sense, if any, can Rayleigh quotient iteration be viewed as Newton's method? A surprising and satisfying by-product of our initial study was that we also were able to provide a definitive answer to the question: In what sense, if any, can inverse iteration and shifted inverse iteration be viewed as Newton's method? Since both authors work in the area of nonlinear programming, it is satisfying that the equivalence that we demonstrate between two-norm normalized Newton's method and two-norm normalized inverse iteration, shifted inverse iteration, and Rayleigh quotient iteration are constructed by turning to three known algorithms from the nonlinear programming literature.

Equivalence between two algorithms from different disciplines can be of value. In this case our equivalences come from the areas of numerical linear algebra and

[*] The material presented in this paper began with discussions between the two authors in 1986 when the first author was teaching a course on linear algebra and the second author was teaching a course on nonlinear optimization in the Department of Computational and Applied Mathematics at Rice University. The material has been the subject of numerous presentations over these past 25 years in the broad applied mathematics community. In the earlier days the talks were entitled "If it is Fast and Effective, it Must be Newton's Method". A talk with this title was presented by the second author at the inaugural David Blackwell-Richard Tapia Mathematics Conference at Cornell University in May of 2000. More recently a talk carrying the title of the present paper was presented by the second author at the Institute for Pure and Applied Mathematics at UCLA in June of 2012 at the Conference Celebration of Tony Chan's 60^{th} Birthday. At that conference ensuing discussions with various attendees, including Siam Review Editorial Board representation, motivated the writing of the present paper. Moreover, it was suggested that the paper not only be mathematically rigorous, but contain appropriate historical content and pursue a somewhat more informal format than the standard terse mathematical paper format. We have attempted to do this.

[†] Professor Emeritus, Department of Computational and Applied Mathematics, Rice University, Houston, Texas 77251-1892 (dennis@rice.edu).

[‡] Professor, Department of Computational and Applied Mathematics, Rice University, Houston, Texas 77251-1892 (rat@rice.edu). This author was supported in part by funds associated with the Maxfield-Oshman Chair in Engineering at Rice University.

nonlinear programming. The equivalences allow us to turn to two somewhat distinct collections of intuition, understanding, and tools. Such activity not only aids in the understanding of the algorithms under discussion, but also can facilitate the design of improved algorithms. In this paper we work exclusively in the finite dimensional vector space R^n. As usual, the various p-norms on R^n will be denoted by $\| \|_p$. When no subscript is used we mean an arbitrary norm. All operator norms are the norms induced by the underlying normed vector spaces. By a unit vector $x \in R^n$ we mean an x such that $\|x\| = 1$. The choice of norm will be clear from the context, if not it will be stated. In describing an iterative process we will tend to use integer subscripts on the iterates. However, we will also find it convenient to write the current iterate with no subscript and use the subscript "+" to denote the subsequent iterate. Hence the notation $x_{k+1} = S(x_k)$ or $x_+ = S(x)$ means that x_k or x is the current iterate and x_{k+1} or x_+ is the subsequent iterate.

1.1. Overview of paper. We have relegated technical preliminaries concerning standard convergence notions in computational mathematics to Appendix A. These notions will be important for intelligent discussions concerning subtle, and not so subtle, distinctions in various theories or algorithms discussed in the paper. The reader not familiar with these notions will gain a heightened appreciation for the material in the paper by first becoming familiar with the material in Appendix A. In Section 1.2 we briefly introduce Newton's method and in Section 1.3 we introduce the inverse iteration, shifted inverse iteration, and Rayleigh quotient iteration algorithms for the symmetric eigenvalue problem. A basic premise of this paper is that when an algorithm requires the solution of a square linear system at each iteration and has been "tuned and prune" so that it is effective and fast, then it is highly likely that what resulted is a form of Newton's method. This is discussed in Section 1.4. Three motivational examples of our basic premise are described in Section 1.5. A historical development of inverse iteration is given in Section 2, of shifted inverse iteration in Section 3, and of Rayleigh quotient iteration in Section 4. Several applications of Newton's method for the algebraic eigenvalue problem from the literature are discussed in Section 5; most notably those of Peters and Wilkinson [25] and Tapia and Whitley [33]. In Section 6 we give the historical development of three equality constrained optimization algorithms that form the basis of our normalized Newton's method equivalence results presented in the following three sections. In Section 7, 8, and 9 we demonstrate that inverse iteration, shifted inverse iteration, and Rayleigh quotient iteration respectively can each be viewed as a form of normalized Newton's method. In Section 10 we apply standard Newton's method theory to Rayleigh quotient iteration viewing it as a form of Newton's method in an effort to explain why the algorithm gives cubic convergence and not just quadratic convergence as should be expected. In Section 11 we use our newly gained understanding to derive Rayleigh quotient iteration in a clean and precise alternative manner. Motivated by the impressive role that the 2-norm plays we reflect on the wonderment of the 2-norm in Section 12. Section 13 contains summarizing and concluding statements. Finally references are given in Section 14.

1.2. Newton's method. Consider $F : R^n \to R^n$ and let $x^* \in R^n$ be a zero of f, i.e. $F(x^*) = 0$. Given x_0, Newton's method constructs the sequence

$$x_{k+1} = x_k + \Delta x_k, \quad k = 0, 1, 2, \dots \tag{1.1}$$

by solving the square linear system

$$F'(x_k)\Delta x = -F(x_k) \quad \text{for} \quad \Delta x_k. \tag{1.2}$$

Constructions (1.1) and (1.2) can be concisely written as

$$x_{k+1} = x_k - F'(x_k)^{-1}F(x_k). \tag{1.3}$$

However, numerical analysts prefer the notation (1.1) and (1.2) to (1.3) since the former tacitly implies that a linear system is solved in contrast to constructing a matrix inverse, a more time-consuming task.
Inherent in this formulation is the hope that $\{x_k\}$ will converge to the solution x^*.

Normalized Newton (NN)
For the nonlinear equation problem

$$F(x) = 0,$$

when it is known that $\|x^*\| = 1$, we modify Newton's method as follows

(a) Solve: $\quad F'(x)\Delta x = -F(x)$

(b) Normalize: $\quad x_+ = \dfrac{x + \Delta x}{\|x + \Delta x\|}.$

We shall refer to this modified Newton's method as normalized Newton's method.

1.3. Inverse, shifted inverse, and Rayleigh quotient iteration. Let A be a symmetric $n \times n$ matrix.

The Symmetric Eigenvalue Problem.
Find $(x^* \neq 0, \sigma^*) \in R^n \times R$ such that

$$Ax^* = \sigma^* x^*.$$

The Rayleigh Quotient.

$$\sigma_R(x) = \frac{x^T A x}{x^T x} \quad (x \neq 0).$$

Notice that the Rayleigh quotient is homogeneous of degree zero, i.e., for $x \in R^n$ and $\alpha \in R$ we have

$$\sigma_R(\alpha x) = \sigma_R(x) \quad \alpha \neq 0.$$

The following algorithmic definitions follow those given in Parlett [24].

Inverse Iteration (II)
Choose a unit vector x_0 and for $k = 0, 1, \ldots,$ repeat the following:
(a) Solve $\quad Ay_{k+1} = x_k \quad$ for y_{k+1}
(b) Normalize $\quad x_{k+1} = \dfrac{y_{k+1}}{\|y_{k+1}\|}.$

Shifted Inverse Iteration (SII)
Choose a scalar σ and a unit vector x_0 and for $k = 0, 1, \ldots,$ repeat the following:
(a) Solve $\quad (A - \sigma I)y_{k+1} = x_k \quad$ for $\quad y_{k+1}$

(b) Normalize $\quad x_{k+1} = \dfrac{y_{k+1}}{\|y_{k+1}\|}.$

The scalar σ is called the *shift*.

Rayleigh Quotient Iteration (RQI)
Choose a unit vector x_0 and for $k = 0, 1, \ldots,$ repeat the following:
(a) Solve $\quad (A - \sigma_R(x_k)I)y_{k+1} = x_k \quad$ for $\quad y_{k+1}$

(b) Normalize $\quad x_{k+1} = \dfrac{y_{k+1}}{\|y_{k+1}\|}.$

Most texts do not make a formal distinction between II and SII, since II is the special case of SII where $\sigma = 0$. We prefer to make a distinction because in the corresponding equivalent Newton's method formulations that we present in Sections 7 and 8 in terms of development, application, and general properties there are considerable distinctions.

1.4. Our basic premise. A fundamental observation is that computers can only do arithmetic. Hence the only equations that can be solved on a computer are linear systems, i.e. systems of the form $Ax = b$. If A is square, i.e., the number of unknowns is equal to the number of equations and nonsingular, then the linear system has a unique solution and the problem of solving $Ax = b$ is well-defined. Hence, the activity of using a computer to approximate a solution to a given problem must consist of sequentially solving linear systems. As we have seen Newton's method reduces the solution of a square system of nonlinear equations to the solution of a sequence of square systems of linear equations. It should therefore not be a surprise that a fundamental activity in the computational sciences and applied science and engineering is formulating and applying Newton's method to the problem at hand. However, our concern in this study is the opposite direction as we now describe.

Our Basic Newton's Method Premise.
Most effective and fast iterative procedures that require the solution of a linear system at each iteration are, perhaps very disguised, forms of Newton's method. Moreover this is particularly true of algorithms that have resulted from a tuning and pruning process.

The Newton's Method Identification Process.
Given an effective and fast iterative procedure that requires the solution of a linear system at each iteration and is not known to be Newton's method, demonstrate that it is a form of Newton's method by exhibiting the fundamental underlying, often well-hidden, nonlinear equation.

Our primary application of the Newton's Method Identification Process will be Rayleigh quotient iteration (RQI). As we will demonstrate in Section 4 it is the result of tuning and pruning processes that have been proposed over an extended 81 year period of time. RQI is known to be effective and fast and requires the solution of a linear system at each iteration. In the literature equivalences of RQI to Newton's method are often implied, tacitly assumed, or are somewhat incomplete equivalences. Hence for many years the stage has been set for our contribution.

On the other hand, since inverse and shifted inverse iteration, while effective, are not fast (only linearly convergent) and do not require the solution of a linear system at each iteration, it is quite surprising that we are able to demonstrate their equivalence to a form of normalized Newton's method. This equivalence begs certain questions concerning the lack of the need to solve a linear system at each iteration and the fact that only linear convergence is obtained. We will address these questions later on in the paper. As we shall see these two attributes are not only intimately related but are in direct opposition to each other.

1.5. Three motivational examples. In this section we illustrate the Newton's Method Identification Process with three examples from classical computational methods history.

1.5.1. The Babylonian's algorithm for calculating $\sqrt{2}$. In telling this story we take the usual liberties afforded the story teller concerning complete accuracy.

So, pretend that you are a Babylonian and you want to find a real number x such that

$$x^2 = 2. \tag{1.4}$$

You stare at (1.4) for considerable time and realizing that you can only perform arithmetic you write (1.4) in the equivalent form.

$$x = \frac{2}{x}. \tag{1.5}$$

After more staring you observe that (1.5) suggests the iterative process

$$x_{k+1} = \frac{2}{x_k}, \quad k = 0, 1, 2, \ldots \tag{1.6}$$

You happily try (1.6) with x_0 and obtain $x_1 = \frac{2}{x_0}$, $x_2 = x_0$, $x_3 = \frac{2}{x_0}$, $x_4 = x_0, \ldots$ You realize that for any x_0 which does not satisfy (1.5) you will oscillate between x_0 and $\frac{2}{x_0}$. Not detoured by this failure you return to (1.5) and do more staring, with the intend of making a change. You know that whatever you do to the left-hand side of the equation you must also do to the right-hand side. Adding a constant to both sides does nothing. So you do the next obvious thing and add x to both sides, and simplify. This leads to

$$x = \frac{1}{2}\left(x + \frac{2}{x}\right). \tag{1.7}$$

This tuning and pruning process has led to the iterative process

$$x_{k+1} = \frac{1}{2}\left(x_k + \frac{2}{x_k}\right) \quad k = 0, 1, 2, \ldots \tag{1.8}$$

Now applying (1.8) with $x_0 = 1$ we obtain $x_1 = 1.50000$, $x_2 = 1.41666$, $x_3 = 1.41421$, and $x_4 = 1.41421$. The resulting algorithm is remarkably effective and fast. It is a simple matter to demonstrate that (1.8) is Newton's method applied to the nonlinear function $F(x) = x^2 - 2$.

A common derivation of the Babylonian algorithm (1.8) is the following. A clever Babylonian realized that if x was an underestimate (overestimate) of $\sqrt{2}$, then $2/x$ would be an overestimate (underestimate) of $\sqrt{2}$. Hence the average should be a better estimate, and this leads directly to the iterative procedure (1.8). We prefer the former derivation since it actually utilized the tuning and pruning approach that we have described. However, both derivations can be questioned since the Babylonians did not know algebra at the level described above and they did not iterate their approximation algorithm. See Papakonstantinou and Tapia [23] for more details. Yet, this illustration of our basic premise is satisfying since it occurred nearly 2000 years before the birth of Newton's method.

1.5.2. Unconstrained Optimization. The unconstrained optimization problem in R^n is

$$\min_{x\in R^n} f(x) \tag{1.9}$$

where $f : R^n \to R$. Historically, an algorithm for approximating a solution to this problem has been the formation of the sequence

$$x_{k+1} = x_k + \Delta x_k, \quad k = 0, 1, 2, \ldots \tag{1.10}$$

where Δx_k solves the quadratic program

$$\min_{\Delta x} \ \nabla f(x_k)^T \Delta x + \frac{1}{2}\Delta x^T \nabla^2 f(x_k)\Delta x. \tag{1.11}$$

Observe that essentially what this algorithm does is sequentially minimize a second-order Taylor series approximation of f at x_k ignoring the redundant constant term. The algorithm is effective and fast and known to be equivalent to Newton's method on the square nonlinear system of first-order necessary conditions.

$$F(x) \equiv \nabla f(x) = 0.$$

1.5.3. Equality Constrained Optimization.. The equality constrained minimization problem in R^n is

$$\begin{aligned} \min_{x\in R^n} \quad & f(x) \\ \text{subject to} \quad & h(x) = 0 \end{aligned} \tag{1.12}$$

where $f : R^n \to R$ and $h : R^n \to R^m$ with $m \le n$.

As a first idea we consider combining the Newton nonlinear equation approach with the constrained minimization approach (1.10)-(1.11) to form the algorithm.

$$x_{k+1} = x_k + \Delta x_k, \quad k = 0, 1, 2, \ldots \tag{1.13}$$

where Δx_k solves the quadratic program

$$\begin{aligned} \min_{\Delta x} \quad & \nabla f(x_k)^T \Delta x + \frac{1}{2}\Delta x^T \nabla_x^2 f(x_k)\Delta x \\ \text{subject to} \quad & \nabla h_i(x_k)^T \Delta x + h_i(x_k) = 0, \quad i = 1, \ldots, m. \end{aligned} \tag{1.14}$$

It should be clear that problem (1.14) can be reduced to the linear system representing its first-order necessary conditions. While this approach can be found in the 1960's literature, researchers soon realized that there was no theory to demonstrate that the algorithm was either locally convergent or gave fast convergence when the generated sequence $\{x_k\}$ converged. Along this line, sometime ago, the authors asked then Rice undergraduate student Thomas Hyer to perform extensive numerical experimentation with this algorithm. Hyer validated the conjecture that the algorithm possessed neither local convergence nor fast convergence.

The situation was ripe for a tuning and pruning process to surface. Indeed optimization researchers, circa 1975, developed the intuitive feeling that "constraint curvature", i.e. Hessian information from the constraints, needed to be added to the model subproblem (1.14). This concern led to the following so-called successive quadratic programming method for equality constrained minimization (1.12).

Given (x_0, y_0) for $k = 0, 1, \ldots$ form the sequence $\{(x_k, y_k)\}$ with

$$x_{k+1} = x_k + \Delta x_k \tag{1.15}$$

where Δx_k solves the quadratic program

$$\begin{aligned} \min \quad & \nabla f(x_k)^T \Delta x + \frac{1}{2} \Delta x^T \nabla_x^2 \ell(x_k, y_k) \Delta x \\ \text{subject to} \quad & \nabla h_i(x_k)^T \Delta x + h_i(x_k) = 0, \quad i = 1, \ldots, m. \end{aligned} \tag{1.16}$$

and $y_{k+1} \in R^m$ is the vector of multipliers for the m constraints in the solution of problem (1.16). Here $\nabla_x^2 \ell(x, y)$ is the Hessian with respect to x of the Lagrangian function

$$\ell(x, y) = f(x) + y^T h(x). \tag{1.17}$$

Numerical experience showed that this SQP algorithm was effective and fast. Moreover, as an equality constrained quadratic program it can be represented by a square linear system of equations. Researchers in the area of computational optimization quickly observed the following fact, see Tapia [34] for example.

Theorem 1.1. *The SQP algorithm described by (1.15)-(1.16) is equivalent to Newton's method on the square nonlinear system of first-order necessary conditions for problem (1.12)*

$$\begin{aligned} \nabla_x \ell(x, y) &= 0 \\ h(x) &= 0. \end{aligned} \tag{1.18}$$

Proof. The proof of this fact is straightforward. □

The successive quadratic programming approach that we have described was actually developed for the full nonlinear programming problem, i.e., problem (1.12) with the addition of inequality constraints, see Garcia-Palomares and Mangasarian [10], Han [12], and Powell [27]. However, in this generality the resulting quadratic program analogous to (1.16) has inequality constraints and as such can not be represented by a square linear system of equations. So it is an abuse of terminology to call the SQP approach in this generality a form of Newton's method. It is the restriction of the optimization problem to only equality constraints that well demonstrates our basic Newton's method premise.

Enhanced understanding of the SQP approach (1.15)-(1.17) allowed us to present this algorithm in the equivalent and more intuitive form.

Given (x_0, y_0) for $k = 0, 1, \ldots$, form the sequence $\{(x_k, y_k)\}$ with

$$x_{k+1} = x_k + \Delta x_k$$
$$y_{k+1} = y_k + \Delta y_k$$

where Δx_k with associated multipliers Δy_k solves the quadratic program

$$\min_{\Delta x} \quad \ell(x_k, y_k) + \nabla_x \ell(x_k, y_k)\Delta x + \frac{1}{2}\Delta x^T \nabla_x^2 \ell(x_k, y_k)\Delta x \tag{1.19}$$

$$\text{subject to} \ \ \nabla h_i(x_k)^T \Delta x + h_i(x) = 0, \quad i = 1, \ldots, m. \tag{1.20}$$

The formulation (1.19)-(1.20) highlights the fact that the correct approach to Newton's method for problem (1.12) should be to minimize the quadratic second-order Taylor approximation of the Lagrangian subject to a linearization of the constraints (first-order Taylor approximation).

Later on it was learned that a form of the SQP method had been previously proposed by Wilson [37] in 1963. The following qualifier concerning the history of optimization is probably in order. It is a reasonable task to catalog optimization contributions made post World War II. However, it is impossible to identify all of the contributions made prior to World War II for many reasons including the fact that this history represents 300 years of contributions in the calculus of variations activity. In this activity finite dimensional results were rarely published yet well-understood, as was the case for William Karush's contribution to the Karush-Kuhn-Tucker first-order necessary conditions for nonlinear programming.

We now look forward to demonstrating, with enhanced excitement, not only that Rayleigh quotient iteration is a form of normalized Newton's method, but that the final algorithm represents the quintessential tuning and pruning process.

2. The historical development of inverse iteration. Before we begin with the historical journeys presented in this section and in the next two sections we add to the qualifier made at the end of Section 1. The task of historical development is always daunting. In the present situation this is so because the body of engineering and applied science literature that deals with some aspect of the algebraic eigenvalue problem that has been produced since Lord Rayleigh's seminal work of 1877 is so immense that it has not been possible for computational scientists to stay abreast of all computational related contributions from this literature. While we have tried to present a representative number of steps in the development processes of our algorithms, some works from the literature have undoubtedly been overlooked. We apologize in advance for these shortcomings.

Clearly inverse iteration grew out of the power method for the algebraic eigenvalue problem.

Power Method.
Choose a unit vector x_0 and for $k = 0, 1, \ldots$, repeat the following:

(a) Form $\quad y_{k+1} = Ax_k$

(b) Normalize $\quad x_{k+1} = \dfrac{y_{k+1}}{\|y_{k+1}\|}$.

The Power Method is believed to have begun with Müntz [19] in 1913. Since the Power Method generates a sequence of approximate eigenvectors converging to an eigenvector associated with the largest (in absolute value) eigenvalue, it seems to be a rather direct thought that if one wants an eigenvector corresponding to the smallest (in absolute value) eigenvalue then the Inverse Power Method (inverse iteration) would be created. The popular web source Wikipedia credits Ernst Pohlhausen [26] with the original formulation of inverse iteration in 1921. A quick look at that paper supports the statements made above, concerning creating the inverse power method to approximate the smallest eigenvalue.

3. The historical development of shifted inverse iteration. The contemporary numerical linear algebra community, rather uniformly, attributes shifted inverse iteration to Helmut Wielandt. He presented it in 1944 in an unpublished report [35] written in German. Yet, Peters and Wilkinson in the Introduction of their paper [25] write "[shifted] inverse iteration is usually attributed to Wielandt (1944) though a number of people seem to have had the idea independently." Indeed, Wilkinson [36] himself suggested the algorithm in 1958.

Our sense is that while Pohlhausen may have presented inverse iteration in 1923 his contribution was not recognized and consequently did not impact the computationally minded algebraic eigenvalue community in the way that Wielandt's paper impacted this community. Ipsen [15], [16], writes extensively on Helmut Wielandt and Jim Wilkinson and compares Wielandt favorably with the recognized giant Wilkinson.

Wielandt's presentation of his two algorithms is a bit awkward and confusing. However, in an effort to retain the sense of his contributions we will follow his presentation rather closely, except for notation.

For $x \in R^n$ Wielandt uses the notation x_j to denote the j-th component of x. We will find it more convenient to use the notation $(x)_j$ for this quantity.

Wielandt Shifted Inverse Iteration.
Consider a matrix $A \in R^{n \times n}$ and choose an integer $j \in \{1, \ldots, n\}$, an approximate eigenvalue $\sigma \in R$, and an approximate eigenvector x_0 satisfying $(x_0)_j = 1$. For $k = 0, 1, \ldots$ repeat the following

$$\text{Solve} \quad (A - \sigma I) z_k = x_k \quad \text{for} \quad z_k \tag{3.1}$$

$$\text{Normalize} \quad x_{k+1} = \frac{z_k}{(z_k)_j}. \tag{3.2}$$

Wielandt's normalization (3.2) is commonly used in the algorithmic eigenvalue problem literature and is viewed as allowing for an infinity norm normalization by choosing j as the index of the largest (in absolute value) component.

Wielandt immediately follows his algorithmic presentation by saying that the quantity

$$\sigma + \frac{1}{(z_k)_j} \tag{3.3}$$

gives an estimate that should converge to the true eigenvalue.

After some discussion concerning algorithmic convergence, Wielandt presents one and only one numerical example. In this example the approximate eigenvalue estimate σ in (3.1) changes from iteration to iteration. His algorithm is as follows.

Wielandt Iteration.
Consider a matrix $A \in R^{n\times n}$, an approximate eigenvalue σ_0, an integer $j \in \{1,\dots,n\}$, and an approximate eigenvector x_0 satisfying $(x_0)_j = 1$. For $k = 0, 1, \dots$ repeat the following

(3.4) Solve $(A - \sigma_k I)y_{k+1} = x_k$ for y_{k+1}

(3.5) Normalize $$x_{k+1} = \frac{y_{k+1}}{(y_{k+1})_j}$$

(3.6) Update the eigenvalue estimate
$$\sigma_{k+1} = \sigma_k + \frac{1}{(y_{k+1})_j}.$$

Wielandt iteration (3.4)-(3.6) for the nonsymmetric 3×3 example given in the paper converges to an eigenpair extremely fast. The convergence is clearly quadratic and probably cubic. In accordance with the Section 1.4 basic Newton's method premise of this paper, this observation raised the obvious question. In an attempt to better understand Wielandt iteration, the authors imposed on then Rice applied math graduate student Josef Sifuentes to perform a numerical study centered on comparing Wielandt iteration with Rayleigh quotient iteration. Josef graciously obliged and conducted an extensive study working with matrices of varying size. He considered both symmetric and nonsymmetric matrices, but in the latter case he made sure that the matrices had real eigenvalues. In all cases, except for the nonsymmetric 3 × 3 example presented in Wielandt's paper, RQI outperformed Wielandt iteration. Moreover, in all cases Wielandt iteration gave quadratic convergence. In the case of the example from the paper, the convergence appeared cubic, and the convergence was slightly better than that given by RQI. In several cases for nonsymmetric matrices the convergence of RQI deteriorated to quadratic, but still performed as well as Wielandt iteration.

This numerical study motivated us to conjecture and prove the following theorem. It seems not to be known, and is satisfying in that it is yet another example of our basic premise.

We will find it convenient to write $(x)_j$ as $e_j^T x$ where e_j is the j-th natural basis vector that has 1 as the j-th component and zeros elsewhere.

THEOREM 3.1. *Wielandt iteration (3.4)-(3.6) is equivalent to Newton's method on the nonlinear system*

$$(A - \lambda I)x = 0 \tag{3.7}$$
$$e_j^T x = 1. \tag{3.8}$$

Proof. Newton's method on (3.7)-(3.8) consists of given the pair (x, λ) with $e_j^T x = 1$ constructing $(x + \Delta x, \lambda + \Delta\lambda)$ by solving the linear system

$$\begin{pmatrix} A - \lambda I & -x \\ e_j^T & 0 \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta\lambda \end{pmatrix} = - \begin{pmatrix} (A - \lambda I)x \\ 0 \end{pmatrix}. \tag{3.9}$$

The system (3.9) tell us that

$$x + \Delta x = \Delta\lambda (A - \lambda I)^{-1} x \tag{3.10}$$
$$e_j^T \Delta x = 0 \tag{3.11}$$

From (3.10) using (3.11) and the fact that $e_j^T x = 1$ we obtain

$$\Delta\lambda = \frac{1}{e_j^T (A - \lambda I)^{-1} x}. \tag{3.12}$$

Now, (3.5) follows from (3.10) and (3.12) and (3.6) follows from (3.12). □

Helmut Wielandt was a pure mathematician first and foremost. His first and last love was group theory and in particular permutation groups. In 1942 Wielandt, clearly due to the war effort, was attached to the Kaiser Wilhelm Institute and the Aerodynamics Research Institute at Göttingen; a situation not uncommon during those times for many pure mathematicians in both Germany and the United States. Concerning that period Wielandt writes

> *"I had to work on vibrations problems. I am indebted to that tie for valuable discoveries: on the one hand the applicability of abstract tools to the solution of concrete problems, on the other hand, the - for a pure mathematician- unexpected difficulty and unaccustomed responsibility of numerical evaluation. It was a matter of estimating eigenvalues of non-self-adjoint differential equations and matrices."*[1]

Concerning numerical methods Wielandt was somewhat of a stranger in a strange land. This leads us to ponder the following thoughts:

- Was Wielandt familiar with the Rayleigh quotient? If he had used it in his algorithm he would have been the discoverer of Rayleigh quotient iteration some 13 years prior to Ostrowski as described in the next section. We believe that he was not. Our foundation for this belief comes from the numerical experiments performed by Josef Sifuentes comparing Wielandt iteration to RQI. Wielandt was a scientist, had he seen that RQI was superior to his algorithm and cost no more to implement, he would have presented RQI in his paper.
- Did Wielandt conjecture that his iterative procedure was Newton's method? Again we think not. While he was undoubtedly aware of Newton's method, we believe that it was not in his bag of favorite mathematical tools. Hence we are pleased to present Wielandt iteration as yet another example of the basic premise of this paper.

4. The historical development of Rayleigh quotient iteration.

We present several contributions from the literature that we believe played a role in the development of Rayleigh quotient iteration. In all that follows $\sigma_R(x)$ represents the Rayleigh quotient defined by (1.4).

1877 – Lord Rayleigh (John William Strutt)

In 1877 [30] in a pioneering and critically important text entitled The Theory of Sound, Lord Rayleigh, while studying the natural frequencies of vibrating systems, proposed the following algorithm for approximating an eigenpair of a symmetric matrix A

$$(A - \sigma_R(x)I)x_+ = e_1 \tag{4.1}$$

where e_1 denotes the first natural coordinate vector $(1, 0, \ldots, 0)^T$.

1940 – R.V. Southwell

In 1940 R. V. Southwell's (Sir Richard Southwell) influential text entitled Relaxation

[1] From the website www-groups.dcs.st-and.ac.uk/ history/Biographies/Wielandt.html

Methods in Engineering Science was published. Southwell developed and promoted a school of thought referred to as relaxation methods. The relaxation method philosophy influenced the design of algorithms for algebraic eigenvalue problems, throughout the decade of the 1940's and well into the decade of the 1950's. Following Ipsen [14] we extract from the introduction of S.H. Crandall's 1951 paper [5] entitled "Iterative procedures related to relaxation methods for eigenvalue problems" his view of relaxation methods:

> *"A unique feature of relaxation methods is that they are approximate procedures which are not rigidly prescribed in advance. The precise procedure to be followed is not dictated but is left to the intuition and accumulated experience of the computer. The computer's intelligence is thus an active or dynamic link in the computational chain. It is this fact which has made relaxation so attractive to many computers."*

In those times a computer was a human being, hopefully a creative one.

It is interesting then that students of the Southwell relaxation methods school, in formulating numerical procedures, often intentionally left sufficient flexibility in the formulation for the computer to include numerical techniques and experimentation in the implementation. So relaxation methods were essentially a general framework that allowed flexibility and promoted creativity on the part of both the applied science modeler and the human computer. It is remarkable to see how much change occurred in the ensuing decades concerning the notion of a numerical procedure. Of course the impetus of this change has been the fact that the human computer, with considerable responsibility for making decisions, has been replaced with a digital computer and a programmer who is only not encouraged, but is not allowed to make delicate decisions.

While we do not represent a specific algorithm for Southwell, we felt that it was important to introduce his relaxation methods philosophy since it had a significant influence on the design of many algorithms from this time period in England and in the United States.

1944 – H. Wielandt

We have discussed Wielandt's contribution in detail in the previous section. Clearly, for transparent reasons Wielandt was not influenced by the relaxation method philosophy.

1946 – J. Todd

Following the war in 1946, John Todd returned to King's College in London and gave his first course in numerical mathematics. In 1957 he and his wife, the well respected linear algebraist Olga Taussky Todd, moved to Caltech to start a program in numerical mathematics. John's lecture notes from this extended period were widely circulated. In these notes he presented the following algorithm for the algebraic eigenvalue problem.

$$\text{Approximately Solve} \quad (A - \sigma_R(x)I)x_+ = 0 \quad \text{for} \quad x_+. \tag{4.2}$$

It is abundantly clear that Todd's algorithm (4.2) is given in context of the Southwell relaxation school. In 1992 at a conference at UCLA the second author had the opportunity to ask John what he meant by "approximately solve" since the clear and unique solution is the zero vector. He replied in a pleasant and straightforward manner "approximately solve is to be interpreted by the individual that is designing the algorithm."

1949 – W. Kohn

In a letter of the editor Kohn [17] suggests the following algorithm for the algebraic eigenproblem

$$(A - \sigma_R(x)I)x_+ = e_i \tag{4.3}$$

where e_i, $i = 1, \ldots, n$ is any natural basis vector for R^n. He claims that if x_0 is sufficiently close to an eigenvector, then the sequence $\{\sigma_R(x_k)\}$ converges to the associated eigenvalue and the convergence is quadratic. This quadratic convergence is stated as q-quadratic convergence.

Observe that Kohn's algorithm (4.3) is a trivial modification of (4.1), the algorithm originally proposed by Lord Rayleigh. Kohn did not mention Lord Rayleigh's algorithm or any other algorithm in his brief letter.

1951 – S. Crandall

In 1951 in an interesting paper entitled "Iterative procedures related to relaxation method for eigenvalue problems" and communicated to the Royal Society of London by Sir Richard Southwell himself, Crandall proposes the following algorithm for the algebraic eigenvalue problem

$$(A - \sigma_R(x)I)x_+ = x. \tag{4.4}$$

Actually Crandall considered the more general eigenvalue problem $Ax = \lambda Bx$; however for our comparative purposes it is sufficient to take B equal to the identity matrix.

We first observe that Crandall's algorithm is unnormalized Rayleigh quotient iteration, see Section 1.3.

On page 422 of his paper Crandall derives an expression involving the approximate eigenvector x_k and states

> "The behaviour of this process in the large is simple. The ordinary behaviour of the method results in sequences of λ_k and x_k which approach respectively an eigenvalue λ_p and its corresponding mode, x_p. Thus if x_k differs from x_p by terms of order ϵ, x_{k+1} differs from a multiple of x_p by terms of order ϵ^3. Their Rayleigh quotients have errors of order ϵ^2 and ϵ^6 respectively."

Summarizing, our interpretation of Crandall is that he claims the following:
If

$$\|x_k - x_p\| = O(\epsilon),$$

then there exists a constant C_k so that

$$\|x_{k+1} - C_k x_p\| = O(\epsilon^3), \tag{4.5}$$

and it follows that

$$\|\lambda_k - \lambda_p\| = O(\epsilon^2)$$

and

$$\|\lambda_{k+1} - \lambda_p\| = O(\epsilon^6),$$

where $\lambda_k = \sigma_k(x_k)$ and $\lambda_{k+1} = \sigma_k(x_{k+1})$.

At this point Crandall believes that he has established (what we would call) r-cubic convergence in both the approximate eigenvector sequence and the approximate eigenvalue sequence. However, it is not at all clear what the constant C_k in (4.5) masks concerning these conclusions. This issue will be further addressed in the following section.

In further reinforcing Southwell's influence we point out that the final paragraph of Crandall's paper includes the statement

> *" From the preceding analysis it is possible to infer the following guiding principles to aid the relaxation computer in choosing his tactics."*

1957–1958–A.M. Ostrowski

Ostrowski authored a two-part paper entitled "On the convergence of the Rayleigh quotient iteration for the computation of the characteristic roots and vectors." We first observe that at the time of Ostrowski's paper Rayleigh quotient iteration had not been defined. Hence we must conclude that in this paper he intended to define such an algorithm. Part I [21] of Ostrowski's paper appeared in 1957, and part II [22] appeared in 1958. We will see that motivated by interesting circumstances Ostrowski defines Rayleigh quotient iteration as we know it today in part II of his paper. We now discuss part I of the paper.

Ostrowski–Part I.

Ostrowski begins his paper by writing down (4.2) and saying that his attention was drawn to this method by John Todd who used it in his lectures as long ago as 1945. He followed by saying that the algorithm appears to converge fairly well in numerical practice. We find this statement very uncharacteristic of Ostrowski. For Ostrowski was known to be an excellent mathematician and quite meticulous and precise in his research writings. Yet he makes this statement about a notion that is vague and certainly not well defined. However he recovers quickly by following with the statement:

> *"The crucial point in the discussion above is of course a suitable rule for the computation of the "approximate solution" x_k in (4.2). The rule that I will use in the first part of this discussion consists in taking an arbitrary vector $\eta \neq 0$ and in putting*
>
> $$x_{k+1} = (A - \sigma_R(x)I)^{-1}\eta. \tag{4.6}$$
>
> *The theoretical arguments in support of this rule are given in another paper."*

He then establishes local and q-quadratic convergence for the approximate eigenvalue sequence. In a footnote he mentions that this result is consistent with the note of Kohn.

In Section 9 of the paper Ostrowski expresses concern that the vector η in the right-hand side of (4.6) is fixed. This leads him to combine his idea with the idea of Wielandt's shifted inverse iteration (3.1) to obtain the algorithm

$$(A - \sigma_R(x)I)x_+ = x. \tag{4.7}$$

Let (x^*, σ^*) be an eigenpair of A. Ostrowski proves local (x_0 near x^*) convergence for the approximate eigenvalue sequence $\{\lambda_k = \sigma_R(x_k)\}$ to σ^* with the property

$$\frac{(\lambda_{k+1} - \sigma^*)}{(\lambda_k - \sigma*)^3} \to \gamma \quad \text{as} \quad k \to \infty \tag{4.8}$$

for some positive γ. Observe that (4.8) reflects q-cubic convergence.

Now, Ostrowski was aware of Kohn [17], but he was not aware of Crandall [5]. Notice that Crandall's algorithm (4.4) is exactly Ostrowski's algorithm (4.7). While Ostrowski's paper was in print George Forsythe, founder of the Computer Science Department at Stanford and close colleague, directed him to Crandall's work. Ostrowski added a note to the paper, while in print, acknowledging that Forsythe had directed him to Crandall's paper. He then added the statement:"Professor Crandall establishes the cubic character of convergence of x_k in the rule (4.7). However, he does not arrive at our asymptotic formula (4.8), which is the principle result of our paper."

Ostrowski-Part II

Ostrowski begins part II of his paper reflecting back on Crandall's paper. He writes

> *"In the paper by Crandall the cubic character of the convergence is proved in the following sense for the sequence of vectors x_k obtained by the rule (4.7):*
> *If $x_k \to \eta$, and if for a sequence of numbers ϵ_k with $\epsilon_k \to 0$,*
>
> $$x_k - \eta = O(\epsilon_k),$$
>
> *then*
>
> $$x_{k+1} - \eta = O(\epsilon_k^3).$$
>
> *However, from this result it does not follow that $\lambda_k - \sigma = O((\lambda_k - \sigma)^3)$, and still less that the asymptotic formula (4.8) holds."*

What Ostrowski seems to be implying is that the r-cubic convergence of the approximate eigenvector sequence $\{x_k\}$ does not imply the q-cubic convergence of the approximate eigenvalue sequence $\{\lambda_k\}$. Ostrowski seems to have missed the fact that Crandall claimed r-cubic convergence of the approximate eigenvalue sequence $\{\lambda_k\}$; but this also would not imply the q-cubic convergence of this sequence that Ostrowski established.

We now express two concerns of our own with Crandall's conclusions. The first is that in the presence of the troublesome constance C_k in (4.5) we do not subscribe to Crandall's conclusions of cubic convergence for the x_k sequence and the λ_k sequence. However, in some sense this is academic since the following result is our main objection to Crandall's conclusions.

THEOREM 4.1. *The iteration sequence $\{x_k\}$ generated by Crandall-Ostrowski iteration (4.7) can not converge.*

Proof. Suppose $x_k \to \eta$. Then from (4.7) we have

$$A\eta = (1 + \sigma_R(\eta))\eta.$$

Hence η is an eigenvector for A with corresponding eigenvalue $1 + \sigma_k(\eta)$ but this means

$$1 + \sigma_R(\eta) = \sigma_R(\eta)$$

and leads to a contradiction. □

While the approximate eigenvector sequence will not converge it is certainly possible that the approximate eigenvalue sequence will converge. In Section 21 of his paper Ostrowski demonstrates that if the x_k iterates in (4.7) are normalized using the 2-norm, and the approximate eigenvalue sequence converges to an eigenvalue, then the normalized approximate eigenvector sequence will converge to the corresponding eigenvector.

We have finally reached our destination starting with Lord Rayleigh in 1877 and ending with Ostrowski in 1958. We have completed an 81 year journey of tuning and pruning that has taken us to Rayleigh quotient iteration as we know it today.

5. The eigenvalue problem and Newton's method. The literature contains attempts to obtain an eigenvalue of a real symmetric matrix by minimizing the Rayleigh quotient. However, such an activity is not productive when employing algorithms related to Newton's method. This is because Newton's method does not behave well when the sought solution is not isolated. An attempt at correcting this is to minimize the Rayleigh quotient subject to the constraint that a solution must have a 2-norm equal to one. In this case the solutions will be isolated for simple eigenvalues. This leads to what we call

The Eigenvalue Optimization Problem
Given a real symmetric matrix $A \in \mathcal{R}^{n\times n}$

$$\begin{aligned} &\text{minimize} \quad x^TAx \\ &\text{subject to } x^Tx - 1 = 0. \end{aligned} \tag{5.1}$$

In deriving the first-order necessary conditions for problem (5.1) we first consider the Lagrangian

$$\ell(x,\lambda) = x^TAx - \lambda(x^Tx - 1) \tag{5.2}$$

and then write

$$\nabla\ell(x,\lambda) = 0,$$

which gives us our first-order necessary conditions as

$$(A - \lambda I)x = 0 \tag{5.3}$$

$$x^Tx - 1 = 0. \tag{5.4}$$

Observe that the unit eigenvectors of A are stationary points of the Lagrangian and the corresponding eigenvalues are the Lagrange multipliers. Also observe that the nonlinear system (5.3)-(5.4) is square with isolated solutions for simple eigenvalues; hence it is amenable to Newton's method.

Anselone and Rall [1] in 1967 consider applying Newton's method to the algebraic eigenvalue problem by applying Newton's method to the square nonlinear system

$$(A - \lambda I)x = 0 \tag{5.5}$$

$$\phi(x) - 1 = 0. \tag{5.6}$$

They call (5.6) the normalization equation. The choice of $\phi(x) = x^Tx$, i.e., normalizing with the 2-norm squared is a very popular choice.

Anselone and Rall list a host of authors who have employed (5.5)-(5.6) with the 2-norm squared normalization dating back to the celebrated L.V. Kantorovich as early as 1948. Many of these applications were in infinite dimensional Hilbert spaces.

It seems quite natural to extend the Anselone-Rall platform for Newton's method to a platform for normalized Newton's method by defining the normalized Newton iterate as

$$x_+ = \frac{x + \Delta x}{\phi(x + \Delta x)} \tag{5.7}$$

where $x + \Delta x$ is the Newton iterate obtained from the nonlinear system (5.5)-(5.6) and ϕ is the normalization function used in (5.6).

Concern for the fact that shifted inverse iteration may require the solution of nearly singular linear systems led Peters and Wilkinson [25] in a paper entitled "Inverse iteration, ill-conditioned equations and Newton's method" to study classical inverse iteration in the context of normalized Newton's method. They naturally first turned to Newton's method on the nonlinear system (5.3)-(5.4) followed by a 2-norm normalization. However, they quickly realized that if instead they used the linear normalization constraint, $e_j^T x = 1$ in (5.6) for some j, then, since Newton iterates satisfy linear constraints, normalized Newton's method would reduce to Newton's method. Hence there would be no need for the normalization step (5.7) and the quadratic convergence of the algorithm would be guaranteed by standard Newton's method theory, unlike the 2-norm case. By choosing j to be the index of the component of x with largest absolute value the normalizing constraint $e_j^T x = 1$ can be viewed as an ∞-norm normalization. Hence in what follows we will slightly abuse terminology and refer to it as the ∞-norm normalization.

In comparing Newton's method on (5.3)-(5.4) to shifted inverse iteration Peters and Wilkinson made the following statement.

> *One step of inverse iteration gives the same result to first order as one Newton step. It is interesting though, that inverse iteration is sometimes the exact equivalent of the classical Newton's method. This is true, for example for the system*

$$\begin{aligned} Ax - \lambda x &= 0 && (5.8)\\ e_j^T x &= 1." && (5.9) \end{aligned}$$

Often in the literature we see authors state that Peters and Wilkinson [25] demonstrated that shifted inverse iteration is Newton's method to first order . The statement, without further explanation, is meaningless. Therefore we first give a formal statement of the Peters-Wilkinson understanding.

THEOREM 5.1. *The eigenvector estimate obtained from one step of Newton's method of the system (5.3)-(5.4) with initial estimates (x_0, λ_0) is a scalar multiple of the eigenvector estimate obtained from one step of shifted inverse iteration (3.1)-(3.2) with initial estimate $(x_0, \sigma = \lambda_0)$. Moreover, if instead we take a Newton step on the system (5.8)-(5.9), then the two estimates are identical.*

Proof. The proof is essentially the same as the proof of Theorem 3.1. □

We now collect some of our observation concerning the Peters-Wilkinson study.

OBSERVATION 5.1. *Theorem 5.1 tells us that the proper interpretation of the Peters-Wilkinson statement that "[shifted] inverse iteration is Newton's method to first order" is that the eigenvector estimate obtained from Newton's method on (5.3)-(5.4) is a scalar multiple of that obtained from [shifted] inverse iteration and nothing whatsoever is implied about eigenvalue information.*

Observation 5.2. *We find it interesting that in proposing Wielandt iteration (5.8)-(5.9), which can be viewed as shifted inverse iteration with an eigenvalue estimate update rule, Wielandt merely wrote down the eigenvalue estimate update rule (3.6) without any hint as to where it came from. However, we were able to show that with this rule his algorithm is equivalent to Newton's method on (5.8)-(5.9). On the other hand, seemingly unaware of Wielandt iteration, Peters and Wilkinson proposed Newton's method on (5.8)-(5.9). They used simple algebra to identify the corresponding eigenvalue estimate update rule and combined it with shifted inverse iteration to obtain Wielandt iteration in exactly the form given by Wielandt. So in posing their Newton's method Peters and Wilkinson rediscovered Wielandt iteration.*

Observation 5.3. *In researching material for the present paper the second author became intrigued with the Peters-Wilkinson quick dismissal of 2-norm normalized Newton's method in favor of ∞-norm normalized Newton's method in large part because of the latter algorithm reduced to Newton's method, and as such quadratic convergence was guaranteed; while in the former case the retention of quadratic convergence was not at all obvious. Hence he recruited then Rice Computational and Applied Mathematics graduate student David Whitley to do a numerical study of these two approaches. The study showed that while the ∞-norm formulation gave quadratic convergence as expected, the 2-norm formulations demonstrated a convergence rate that was clearly better than quadratic, but not yet cubic. Whitley did a superb job on this task and played a major role in conjecturing and establishing the following theorem.*

Theorem 5.2 (Tapia-Whitley (1988)). *Let A be a symmetric matrix, x_0 a vector with $\|x_0\|_2 = 1$, and λ_0 a scalar that is not an eigenvalue of A. Also let the sequence $\{x_k, \lambda_k\}$ be generated by the 2-norm normalized Newton's method on the nonlinear system (5.8)-(5.9) with initial data $\{x_0, \lambda_0\}$.*

If $\{\lambda_k\}$ converges to an eigenvalue λ and $\lambda_k \neq \lambda_*$ for all k, then $\{\lambda_k\}$ converges with q-order $1+\sqrt{2}$, and $\{x_k\}$ converges with the same q-order to a corresponding eigenvector x_*.

Tapia and Whitley conclude their study with the following remarks.

> *"The motivation for this work was the intriguing and, to us, surprising nonintegral superquadratic convergence rate indicated by our numerical experiments. We were also quite pleased that we were able to establish a q-rate of convergence of $1+\sqrt{2}$ both for the x-iterate alone and for the λ-iterate alone. The classical Newton's method theory gives q-quadratic convergence in the pair (x, λ) for the Peters-Wilkinson algorithm, which implies an r-rate in each of the x and λ. The direct proof given by Dongarra, Moler, and Wilkinson [7] also only establishes r-quadratic in x for the same algorithm.*
> *We experimented with several problem formulations that used p-norms other than the 2-norm; for no value of p except $p = 2$ did our estimated convergence rates exceed 2. On the basis of these experiments, we consider it extremely unlikely that any p-norm formulation other than the one presented here produces superquadratic convergence."*

6. Historical development of four equality constrained optimization algorithms.

For the sake of convenience we first restate the unconstrained optimization problem (1.1) and the equality constrained optimization problem (1.12) in R^n.

Unconstrained Optimization. The unconstrained optimization problem in R^n is

$$\min_{x \in R^n} f(x) \tag{6.1}$$

where $f : R^n \to R$.

Equality Constrained Optimization. The equality constrained minimization problem R^n is

$$\begin{aligned} \min_{x \in R^n} \quad & f(x) \\ \text{subject to} \quad & h(x) = 0. \end{aligned} \tag{6.2}$$

Throughout the history of optimization, unconstrained optimization has been more receptive to effective theory and algorithmic construction than has equality constrained optimization. Hence, since the 1940's or so there has been considerable activity at addressing the equality constrained optimization problem by instead considering an unconstrained optimization problem or a sequence of such problems. This latter approach carries the flavor of Newton's method in that a nonlinear problem is addressed by considering a sequence of more accessible problems, in the Newton case linear problems. The approach of reducing the solution of a constrained optimization problem to solving a sequence of unconstrained optimization problems became known as sequential unconstrained optimization techniques or SUMT as was highlighted by the very successful 1968 text "Nonlinear Programming: Sequential Unconstrained Minimization Techniques" [8] by Fiacco and McCormick.

Recall that the first-order necessary conditions for the equality constrained optimization problem (6.2) are

$$\begin{aligned} \nabla_x \ell(x, \lambda) &= 0 \\ h(x) \quad &= 0 \end{aligned} \tag{6.3}$$

where the Lagrangian $\ell(x, \lambda)$ is given by

$$\ell(x, \lambda) = f(x) + \lambda^T h(x). \tag{6.4}$$

6.1. The Courant 2-norm penalty function method. In 1943 [4] Courant proposed what today is called the 2-norm penalty function for problem (6.2)

$$P(x; c) = f(x) + \frac{c}{2} h(x)^T h(x) \quad (c > 0). \tag{6.5}$$

The constant c is called the penalty constant. We first observe that

$$\nabla P(x; c) = \nabla f(x) + \nabla h(x) c \, h(x). \tag{6.6}$$

We are interested in obtaining solutions of problem (6.2) by minimizing the penalty function (6.5). Towards that end we observe that a solution x_* of problem (6.2) with associated multiplier λ_* must satisfy the necessary conditions (6.3). Hence if x^* is also a minimizer of the penalty function (6.5) we must have that $\nabla P(x^*, d) = 0$. This implies from (6.6) that

$$c \, h(x^*) = \lambda^*. \tag{6.7}$$

However , in general $\lambda^* \neq 0$ and since $h(x^*) = 0$ formally we must have $c = +\infty$ so that $\infty \cdot 0 = \lambda^*$. Hence, the 2-norm penalty function method consists of sequentially minimizing the penalty function (6.5) for a sequence of penalty constants $\{c_k\}$ such that $c_k \to +\infty$. In spite of its obvious shortcomings this technique has seen fairly wide use in applications throughout the years.

6.2. The Hestenes-Powell augmented Lagrangian multiplier method.. We consider replacing the 2-norm penalty function (6.5) with the augmented Lagrangian for problem (6.2)

$$L(x,\lambda;c) = f(x) + \lambda h(x) + \frac{c}{2}h(x)^T h(x) \quad (c > 0) \tag{6.8}$$

to obtain the augmented Lagrangian multiplier method

Given (x,λ,c) obtain x_+ as the solution of

$$\min_x L(x,\lambda;c).$$

Update

$$\lambda \to \lambda_+$$
$$c \to c_+.$$

Observe that a minimizer in x of the augmented Lagrangian (6.8) must satisfy the necessary condition $\nabla_x L(x,\lambda;c) = 0$ where

$$\nabla_x L(x,\lambda;c) = \nabla f(x) + (\lambda + c\,h(x))\nabla h(x). \tag{6.9}$$

Hence in contrast to the penalty function, minimizers of the Lagrangian are not precluded from being solutions of the constrained problem (6.2) for finite values of c. So in the multiplier method it is not necessary to choose $c \to +\infty$. However, the rub is that if we obtain convergence in the multiplier method, say $(x_k\, lambda_k; c_k) \to (x^*, \lambda^*; c^*)$ it does not follow from (6.9) alone that $h(x^*) = 0$ in other words stationary points of the augmented Lagrangian, or their limits, are not necessarily stationary points of problem (6.2); let alone solutions of problem (6.2).

Hestenes [13] and Powell [29] independently are credited with formulating the multiplier method in 1969. Hestenes' motivation came from the desire to avoid the need to use an unbounded sequence of penalty constants as is required in the penalty function method.

6.3. The Fletcher exact penalty function method. In the interest of efficiency it was natural for researchers in optimization to ask if it is possible to move away from minimizing a sequence of penalty functions to minimizing just one penalty function in order to obtain the solution of the constrained optimization problem (6.2). Such a function has historically been called an exact penalty function. In 1970 Fletcher [9] proposed the following exact penalty function

$$F(x;c) = L(x,\lambda(x);c) \tag{6.10}$$

where $\lambda(x)$ is the well-known multiplier approximation formula

$$\lambda(x) = -(\nabla h(x)^T \nabla h(x))^{-1} \nabla h(x)^T \nabla f(x). \tag{6.11}$$

The multiplier approximation formula is often called the least-squares multiplier approximation formula because it arises of the least-squares solution for λ of the necessary condition

$$\nabla f(x) + \lambda \nabla h(x) = 0.$$

Fletcher demonstrated that for c sufficiently large solutions of problem (6.2) are minimizers of his penalty function (6.10), i.e. $F(x;c)$ is an exact penalty function.

Obvious criticisms of Fletcher's approach are

- How large is sufficiently large for the penalty constant c?
- The derivatives of f and h are out of phase, e.g., $\nabla F(x;c)$ involves $\nabla^2 f(x)$ and $\nabla^2 h_i(x)$.

6.4. The multiplier substitution equation. Tapia [31] suggested dealing with the deficiencies associated with the Fletcher exact penalty function approach described above by instead of working with the unconstrained optimization problem of minimizing the Fletcher exact penalty working with the nonlinear equation problem

$$\nabla_x L(x, \lambda(x); c) = 0 \tag{6.12}$$

where $\lambda(x)$ is given by (6.11). A strong justification of this approach is the following theorem; appropriate counterparts do not hold for the previous two penalty function methods. As is standard in the literature we call a vector x a regular point of problem (6.2) if the set $\{\nabla h(x)_1, \dots, \nabla h_m(x)\}$ is linearly independent.

THEOREM 6.1. *Let the vector x be a regular point of problem (6.2). Then x is a zero of the multiplier substitution equation (6.12) if and only if it is a stationary point of problem (6.2).*

Proof. We first observe that

$$\nabla_x L(x, \lambda(x); c) = \nabla f(x) + \nabla h(x)(\lambda(x) + c\, h(x)). \tag{6.13}$$

Let (x^*, λ^*) be a solution of (6.3). Then $h(x^*) = 0$ and $\lambda(x^*) = \lambda^*$, so x^* is a zero of the multiplier substitution equation (6.12). Now let x^* satisfy (6.12), substitute $\lambda(x^*)$ into (6.12), and left multiply the expression by $\nabla h(x^*)^T$ to obtain

$$\nabla h(x^*)^T \nabla h(x^*)\, c\, h(x^*) = 0. \tag{6.14}$$

Since we are assuming that x^* is a regular point the matrix in (6.14) is invertible. Hence $c\, h(x^*) = 0$, and since we are assuming that $c \neq 0$ we must have $h(x^*) = 0$. So $(x^*, \lambda^* = \lambda(x^*)$ is a solution of (6.3) and x^* is a constrained stationary point. □

7. Inverse iteration as Newton's method.

At the beginning of our journey we offer the reader the following road sign. We will derive theory establishing the equivalence between the 2-norm normalized Newton's method on an appropriate equation described in the previous four sections and inverse, shifted inverse, and Rayleigh quotient iteration. Each of the three equations involves a penalty constant. As the reader will see it is a straightforward matter to show that the Newton iterate is a scalar multiple of the iterate in question. The challenge will be to show that it is possible to choose the penalty constant so that the scalar is positive; so that the 2-norm normalization will reduce it to 1. Instead of -1, and the iterates in question will coincide.

We now present the following theorem showing that 2-norm normalized Newton's method on the penalty function (6.5) for the eigenvalue optimization problem (6.1)

$$P(x, c) = x^T A x + \frac{c}{2}(x^T x - 1)^2 \quad \text{for} \quad c \neq 0 \tag{7.1}$$

is equivalent to 2-norm normalized inverse iteration as defined in Section 1.3.

THEOREM 7.1. *Consider a 2-norm unit vector x_0, a penalty constant $c \neq 0$, and a symmetric matrix A. Assume that the matrices A and $A + c\,x_0 x_0^T$ are invertible. Let x_{II} be the 2-norm inverse iteration iterate obtained from x_0, and let x_{NN} be the 2-norm normalized Newton iterate obtained from the penalty function (7.1) with penalty constant c and initial iterate x_0. Then*

$$x_{NN} = \pm x_{II}. \tag{7.2}$$

Moreover, the plus sign results and the two iterates are identical for the following choices of penalty constant:

$$\begin{array}{lll} c > 0 \quad \textit{or} \quad c < -\dfrac{1}{x_0^T A^{-1} x_0} & \textit{if} & x_0^T A^{-1} x_0 > 0, \\ \qquad c > 0 & \textit{if} & x_0^T A^{-1} x_0 = 0, \\ \textit{and} \quad 0 < c < -\dfrac{1}{x_0^T A^{-1} x_0} & \textit{if} & x_0^T A^{-1} x_0 < 0. \end{array} \tag{7.3}$$

Proof. We have for inverse iteration

$$x_{II} = \frac{A^{-1} x_0}{\|A^{-1} x_0\|_2}. \tag{7.4}$$

Now, for Newton's method we need the following quantities derived from (7.1)

$$\nabla P(x; c) = 2[Ax + c\,(x^T x - 1)x] \tag{7.5}$$

and

$$\nabla^2 P(x; c) = 2[A + c\,xx^T + c\,(x^T x - 1)I]. \tag{7.6}$$

So Δx the Newton correction for the 2-norm unit vector x_0 satisfies

$$[A + c\,x_0 x_0^T]\Delta x = -Ax_0. \tag{7.7}$$

From (7.7) using simple algebraic manipulations we obtain

$$x_{NN} = \frac{-c\,x_0^T Ax}{\left|c\,x_0^T Ax\right|} \frac{A^{-1} x_0}{\|A^{-1} x_0\|_2}. \tag{7.8}$$

Expression (7.2) follow directly from (7.8). Our task now is to determine conditions that guarantee that $c\,x_0^T Ax < 0$, which in turn will guarantee our equality of iterates. Towards this end we first observe that

$$\Delta x = -(A + c\,xx^T)^{-1} Ax. \tag{7.9}$$

We next call on the Sherman-Morrison formula (see Page 50 of Ortega and Rheinboldt [18]) to expand the inverse term in (7.9) and straightforward algebra to obtain

$$-c\,x^T \Delta x = c - \frac{c\,x^T A^{-1} x}{[\frac{1}{c} + x^T A^{-1} x]} \tag{7.10}$$

a reasonable amount of reflection on (7.10) leads to (7.3). □

It is well known that inverse iteration is linearly convergent. Yet we have just demonstrated that it is equivalent to 2-norm normalized Newton's method; an algorithm known to be quadratically convergent (see Theorem A.1 of Appendix A). Hence we must address this apparent inconsistency. Our algorithm is 2-norm normalized Newton's method on the nonlinear equation

$$\nabla P(x; c) = 2[Ax + c\,(x^T x - 1)x] = 0. \tag{7.11}$$

We know that our algorithm converges to a 2-norm unit eigenvector, say x^* with associated eigenvalue λ^*. Hence from (7.11)

$$\nabla P(x^*; c) = \lambda^* \neq 0. \tag{7.12}$$

So the rub is that while the 2-norm normalization undoubtedly enhances convergence, it precludes convergence to a zero of the equation on which we are performing Newton's method; as a result we only have linear convergence. Now, if we removed the normalization step, then we would retain quadratic convergence, but at the expense of having to solve a different linear system at each iteration. This balance is mathematical poetic justice.

It is also worth stressing that the 2-norm normalized Newton iterates, except for sign, are independent of the penalty constant c.

8. Shifted inverse iteration as Newton's method. We present the following theorem showing that 2-norm normalized Newton's method on the 2-norm augmented Lagrangian (6.8) for the eigenvalue optimization problem (6.1)

$$L(x, \lambda; c) = x^T Ax - \lambda(x^T x - 1) + \frac{c}{2}(x^T x - 1)^2 \quad \text{for} \quad c \neq 0 \tag{8.1}$$

is equivalent to 2-norm normalized shifted inverse iteration, with shift λ, as defined in Section 1.3.

Theorem 8.1. *Consider a 2-norm unit vector x_0, a penalty constant $c \neq 0$, a symmetric matrix A, and a scalar λ. Assume that the matrices $B = A - \lambda I$ and $B + c\,x_0 x_0^T$ are invertible. Let x_{SII} be the 2-norm normalized shifted inverse iteration iterate obtained from x_0 with shift λ, and let x_{NN} be the 2-norm normalized Newton iterate obtained from (8.1), the augmented Lagrangian for the eigenvalue optimization problem (6.1) with penalty constant $c \neq 0$, multiplier estimate λ, and initial iterate x_0. Then*

$$x_{NN} = \pm x_{SII}. \tag{8.2}$$

Moreover, the plus sign results and the two iterates are identical for the following choices of penalty constant

$$\begin{aligned} c > 0 \quad \textit{or} \quad c < -\frac{1}{x_0^T B^{-1} x_0} \qquad & \textit{if} \quad x_0^T B^{-1} x_0 > 0, \\ c > 0 \qquad & \textit{if} \quad x_0^T B^{-1} x_0 = 0, \\ \textit{and} \qquad 0 < c < -\frac{1}{x_0^T B^{-1} x_0} \qquad & \textit{if} \quad x_0^T B^{-1} x_0 < 0. \end{aligned} \tag{8.3}$$

Proof. We observe that shifted inverse iteration can be viewed as inverse iteration with the matrix$A$ replaced by the matrix $B = A - \lambda I$. Moreover Newton's method on the augmented Lagrangian can be viewed as Newton's method on the penalty function with the matrix A replaced by the matrix B. Hence our theorem follows from Theorem 7.1 by making this substitution. □

All the comments made at the end of Section 8 concerning inverse iteration apply here to shifted inverse iteration. Reiterating, the 2-norm normalization, undoubtedly enhances global convergence, allows for an implementation that does not require us to solve a new linear system at each iteration, but precludes quadratic convergence. This latter statement follows from the fact that while convergence is obtained, it is not to a zero of the governing Newton's method equation.

9. Rayleigh quotient iteration as Newton's method. As argued in our introductory sections, in attempting to demonstrate that a given algorithm is Newton's method, the challenge most often is the identification of the underlying Newton's method equation. Historically, this has certainly been the case concerning the interpretation of Rayleigh quotient iteration (RQI) as Newton's method. In our current search it is reasonable to consider a function of the form

$$F(x) = (A - \lambda(x)I)x \tag{9.1}$$

were $\lambda(x)$ is a C^1 functional defined on R^n. If x^* is a zero of F then x^* will be an eigenvector of A with corresponding eigenvalue $\lambda(x^*)$. On the other hand we certainly want 2-norm unit eigenvectors, i.e. solutions of (6.3)-(6.4), to be zeros of F in (9.1). An immediate way of accomplishing this objective is to ask $\lambda(x)$ in (9.1) to agree with the Rayleigh quotient for 2-norm unit vectors.

The present study began sometime in the late 1980's when the first author, after a linear algebra class lecture, approached the second author asking "do you think that RQI is normalized Newton's method on the equation $F(x) = 0$ where F is given by (9.1) with $\lambda(x)$ equal to the Rayleigh quotient." A quick calculation showed that for every initial estimate x, the Newton iterate was the zero vector; hence the normalized Newton iterate was not defined. Motivated by the fact that we could not be further away from the truth, i.e. a complete equivalence, we confidently embarked on the present study.

It follows then that for various choices of the functional λ in (9.1) we will have various levels of equivalence between RQI and normalized Newton's method on (9.1). A necessary condition for the equivalence to be complete is that the set of exceptional points, points where the Newton iterate is zero, is empty.

In the middle 1980's Chatelin [2],[3] produced excellent work on Newton's method and the algebraic eigenvalue problem. She claimed an equivalence between RQI and the normalized Newton's method applied to (9.1) with the choice for $\lambda(x)$ given by

$$\lambda_c(x) = x^T Ax. \tag{9.2}$$

Observe that Chatelin's choice (9.2) coincides with the Rayleigh quotient for 2-norm unit vectors. However, if A is nonsingular and x satisfies $\lambda_c(x) = 0$, then the Newton iterate is zero and the normalized Newton iterate is not defined. To see this, observe that $\nabla\lambda_c(x)^T x = 2\lambda_c(x)$ and $F'(x) = (I - 2xx^T)A - \lambda_c(x) \cdot I$. When $\lambda_c(x) = 0$, the Newton equation becomes

$$(I - 2xx^T)Ax_+^N = 0, \tag{9.3}$$

and since $(I - 2xx^T)$ and A are nonsingular we must have $x_+^N = 0$. Clearly $x = (1, 0)^T$ and

$$A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

are such that $\lambda_c(x) = 0$ and A is nonsingular. Thus, Chatelin's result does not give a complete equivalence.

The multiplier substitution equation (6.12) for the eigenvalue optimization problem (6.1) is written

$$F(x; c) = (A - \lambda_c(x) \cdot I)x = 0 \quad \text{for} \quad c \neq 0 \tag{9.4}$$

where

$$\lambda_c(x) = \lambda(x) - \frac{c}{2}(x^T x - 1) \tag{9.5}$$

with

$$\lambda_R(x) = \frac{x^T A x}{x^T x} \quad x \neq 0. \tag{9.6}$$

It is certainly worth noting that for the eigenvalue optimization problem (6.1) the least-squares multiplier approximation formula (6.11) becomes the Rayleigh quotient. This should not be a complete surprise. We also note that the multiplier substitution function $F(x; c)$ given in (9.4) has the form (9.1) as expected. We now present our equivalence theorem.

THEOREM 9.1. *Consider a 2-norm unit vector x_0, a penalty constant $c \neq 0$, a symmetric matrix A, and a scalar λ. Assume that the matrix $B = A - \lambda_k(x_0)I$ is invertible. Let x_{RQI} be the 2-norm Rayleigh quotient iterate obtained from x_0 and let x_{NN} be the 2-norm normalized Newton iterate obtained from (9.4), the multiplier substitution equation for the eigenvalue optimization problem (6.1) with penalty constant c and initial iterate x_0. Then*

$$x_{NN} = \pm x_{RQI}. \tag{9.7}$$

Moreover, there exist choices for the penalty constant c that lead to the plus sign and therefore the equality of the two iterates.

Proof. We have for 2-norm normalized Rayleigh quotient iteration

$$x_{RQI} = \frac{(A - \lambda_R(x_0)I)^{-1} x_0}{\|(A - \lambda_R(x_0)I)^{-1} x_0\|_2}. \tag{9.8}$$

Now, for Newton's method we need the following quantities derived from (9.4)-(9.6)

$$\nabla \lambda_R(x) = \frac{2(A - \lambda_R(x)I)x}{x^T x} \quad x \neq 0, \tag{9.9}$$

$$\nabla \lambda_c(x) = \nabla \lambda_R(x) - c\, x, \tag{9.10}$$

$$F'(x; c) = (I - 2xx^T)(A - \lambda_R(x)I) + c\, xx^T. \tag{9.11}$$

For convenience we introduce the notation

$$B = A - \lambda_R(x)I \tag{9.12}$$

and

$$D = (I - 2xx^T)B. \tag{9.13}$$

Observe that if x is a 2-norm unit vector and B is invertible then so is D since $(I - 2xx^T)^2 = I$.

Our Newton equation is

$$F'(x; c)\Delta x(c) = -F(x; c) \tag{9.14}$$

writing x for x_0 and appealing to (9.4), (9.5), (9.6), (9.11), (9.12), (9.13) and (9.14) allows us to write the Newton iterate as

$$x + \Delta x(c) = \alpha(c) B^{-1} x \tag{9.15}$$

where

$$\alpha(c) = [2x^T B \Delta x(c) - c\, x^T \Delta x(c)] B^{-1} x. \tag{9.16}$$

So (9.7) follows from (9.8) and (9.15) provided that for $c \neq 0$ we have that $\alpha(c) \neq 0$. This latter fact will follow from our subsequent theorem, Theorem 9.2. Our task is now to show that it is possible to choose c so that $\alpha(c) > 0$ and x_{NN} and x_{RQI} will coincide. Towards this end we first observe that

$$\Delta x(c)] = -(D + c\, x^T x)^{-1} B x. \tag{9.17}$$

As before turning to the Sherman-Morrison formula (see page 5 of [18]) leads to

$$\Delta x(c) = -[D^{-1} - \frac{D^{-1} x x^T D^{-1}}{\frac{1}{c} + x^T D^{-1} x}] BX. \tag{9.18}$$

If follows from (9.18) that $\Delta x(c)$ converges to a vector d^* as $c \to \pm\infty$. Hence, from (9.16) we can force $\alpha(c)$ to $+\infty$ by letting $c \to +\infty$ or $c \to -\infty$. This proves our theorem. □

In sharp contrast to the use of the Rayleigh quotient in (9.1) which always led to a zero Newton iterate and the Chatelin choice which could lead to a zero Newton iterate, we have the following satisfying result.

THEOREM 9.2. *Consider a 2-norm unit vector x and assume that $(A - \lambda_R(x) I)$ is invertible. Then the 2-norm normalized Newton's method for x on the multiplier substitution equation (9.4)-(9.6) gives a complete equivalence with 2-norm normalized RQI for x in the sense that for $c \neq 0$ the Newton iterate is not zero and the 2-norm normalized Newton iterate coincides with either the 2-norm normalized RQI iterate or its negative.*

Proof. From (9.14) we can write

$$(A - \lambda_c(x) I)(x + \Delta x) = (\nabla \lambda_c(x)^T \Delta x) x. \tag{9.19}$$

Now, suppose that $x + \Delta x = 0$. Then since x is a 2-norm with vector $\lambda_c(x) = \lambda_R(x)$ and the coefficient matrix in (9.17) is invertible by assumption. Hence we must have $\nabla \lambda_c(x)^T \Delta x = 0$; hence $\nabla \lambda_c(x)^T x = 0$. But

$$\nabla \lambda_c(x)^T x = c \neq 0.$$

This is the contradiction that establishes the theorem. □

REFERENCES

[1] P.M. ANSELONE AND L.B. RALL, *The solution of characteristic value-vector problems by Newton's method*, Numerische Mathematik 11 (1968), pp. 38–45.

[2] F. CHATELIN, *Simultaneous Newton's iteration for the eigenproblem*, Defect Correction Methods, Computing Supplementum 5, K. Böhmer and H.J. Stetter, eds., Springer Vienna, 1984, pp. 67–74.

[3] F. CHATELIN, *Ill conditioned eigenproblems*, (1986), pp. –.

[4] R. Courant, *Variational methods for the solution of problems of equilibrium and vibrations*, Bulletin of the American Mathematical Society, 49 (1943), pp. 1-23.

[5] S.H. Crandall, *Iterative procedures related to relaxation methods for eigenvalue problems*, Proc. Roy. Soc. London Ser. A (1951), pp. 416–423.

[6] J.E. Dennis Jr. and R.B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall, New York, 1983.

[7] J.J. Dongarra, T.C.B. Moler and J.H. Wilkinson, *Improving the accuracy of computed eigenvalues and eigenvectors*, SIAM J. Numer. Anal. (1983), pp. 23–45.

[8] A.V. Fiacco and G.P McCormick*Nonlinear Programming: Sequential Unconstrained Minimization Techniques* (Wiley, New York, 1968), Reprinted in the SIAM Classics in Mathematics series, Philadelphia, 1990.

[9] R. Fletcher, *A new approach to variable metric algorithms*, Comput. J . 13 (1970), pp. 317–322.

[10] U.M. Garcia-Palomares and O.L. Mangasarian, *Superlinearly convergent quasi-Newton algorithms for nonlinearly constrained optimization problems*, Mathematical Programming 11 (1) (1976), pp. 1–13.

[11] S.-P. Han, *Superlinearly convergent variable metric algorithms for general nonlinear programming problems*, Math. Programming 11 (1976), pp. 263–282.

[12] S.-P. Han, *A globally convergent method for nonlinear programming*, J. Optimization Theory Appl. Math. 22 (1977), pp. 297–309.

[13] M.-R. Hestenes, *Multiplier and gradient methods*, J. Optimization Theory Appl. Math. 4 (1969), pp. 303–320.

[14] I.C.F Ipsen, *Helmut Wielandts contributions to the numerical solution of complex eigenvalue problems*, in Helmut Wielandt, Mathematische Werke, Mathematical Works, 2, Linear Algebra and Analysis, B. Huppert and H. Schneider, eds., Walter de Gruyter, Berlin, 1996, pp. 453-463.

[15] I.C.F. Ipsen, *A history of inverse iteration, in Helmut Wielandt*, Mathematische Werke, Mathematical Works, 2, Linear Algebra and Analysis, B. Huppert, H. Schneider, eds., Walter de Gruyter, Berlin, 1996, pp. 464-472.

[16] I.C.F. Ipsen, *Computing an eigenvector with inverse iteration*, SIAM Review, Society for Industrial and Applied Mathematics 39(2) (1997), pp. 254291.

[17] W. Kohn, *A variational iteration method for solving secular equations*, J. Chem. Phys. 17 (1949), p. 670.

[18] J.M. Ortega and W.C. Rheinboldt, *Iterative solution of nonlinear equations in several variables*, Academic Press, New York, San Francisco, London, 1970.

[19] H. Müntz , *Solution directe de lquation sculaire et de quelques problmes analogues transcendants*, C. R. Acad. Sci. Paris, 156 (1913), pp. 43–46.

[20] H. Müntz , *Sur la solution des quations sculaires et des quations intgrales*, C. R. Acad. Sci. Paris, 156 (1913), pp. 860–862.

[21] A.M. Ostrowski, *On the convergence of the Rayleigh Quotient iteration for the computation of the characteristic roots and vectors*, I, Arch. Rational Mech. Anal. 1 (1957), pp. 233–241.

[22] A.M. Ostrowski, *On the convergence of the Rayleigh Quotient iteration for the computation of the characteristic roots and vectors*, II, Arch. Rational Mech. Anal. 1 (1958), pp. 423–428.

[23] J. Papakonstantinou and R. Tapia, *Origin and evolution of the secant method in one dimension*, Amer. Math. Monthly, 120(6) (2013), pp. 500–518.

[24] B.N. Parlett, *The Symmetric Eigenvalue Problem*, 2nd ed., Classics in Applied Mathematics, 20, SIAM, Philadelphia, 1998.

[25] G. Peters and J.H. Wilkinson, *Inverse iteration, ill-conditioned equations and Newton's method*, SIAM Review, 21 (1979), pp. 339-360.

[26] E. Pohlhausen, *Berechnung der eigenschwingungen statisch-bestimmter fachwerke*, ZAMM - Zeitschrift fr Angewandte Mathematik und Mechanik 1 (1921), pp. 28–42.

[27] M.J.D. Powell, *A fast algorithm for nonlinearly constrained optimization calculations*, In: Numerical Analysis. Proceedings of the Biennial Conference Held at Dundee, June 1977. G.A. Watson, ed., Lecture Notes in Mathematics, 630, Springer, Berlin, Heidelberg, New York, 1978.

[28] M.J.D. Powell, *The convergence of variable metric methods for nonlinearly constrained optimization calculations*, In: Nonlinear programming, 3, O.L. Mangasarian, R.R. Meyer, S.M. Robinson, eds., Academic Press, New York, San Francisco, London, 1978.

[29] M.J.D. Powell, *A method for nonlinear constraints in minimization problems*, In: Optimization, R. Fletcher, ed., Academic Press, New York, NY, 1969, pp. 283298.

[30] J.W.S.B. Rayleigh, *The theory of sound*, MacMillan, 1894.

[31] R.A. Tapia, *Practical Methods of Optimization*, Volume 2: Constrained Optimization, (R.

Fletcher, ed.) SIAM Rev. 26-1 (1984), pp. 143-144.

[32] R.A. Tapia, *On secant updates for use in general constrained optimization*, Mathematics of Computation, 51 (1988), pp. 181203.

[33] R.A. Tapia and D. Whitley, *The projected Newton method has order 1 + sqrt(2) for the symmetric eigenvalue problem*, SIAM J. Numerical Analysis, 25 (1988), pp. 1376–1382.

[34] R.A. Tapia, *Quasi-Newton methods for equality constrained optimization: Equivalence of existing methods and a new implementation*, Nonlinear Programming 3, O. L. Mangasarian, R. R. Meyer, and S. M. Robinson, eds., Academic Press, New York, New York, 1978, pp. 125164, .

[35] H. Wielandt, *Beitra ğe zur mathematischen Behandlung komplexer Eigenwertprobleme*, Teil V: Bestimmung ho herer Eigenwerte durch gebrochene Iteration, Bericht B 44/J/37, Aerodynamische Versuchsanstalt G ottingen, Germany, 1944.

[36] J. Wilkinson, *The calculation of the eigenvectors of codiagonal matrices*, Computer J., 1 (1958), pp. 9096.

[37] R.B. Wilson, *A simplicial algorithm for concave programming*, In: Ph.D. dissertation, Graduate School of Business Administration, Harward University, Boston, 1963.

Appendix A. A brief introduction to convergence rate and algorithmic convergence. When studying iterative algorithms, much can be learned by determining the convergence rate of the sequences that they generate. We strongly emphasize that convergence rate is only a part of the story and does not give the complete picture. As the reader will see from our definitions, convergence rate is essentially an asymptotic property. We only know the behavior of the iteration sequence after some fixed index, i.e., the tail-end of the sequence. In practice, whether this fixed index appears relatively early or relatively late will be more important in determining the effectiveness of the algorithm than will be the convergence rate. However, convergence rate is of value and worth determining whenever possible.

A.1. Convergence rate. Consider a sequence $\{x_k\}\subset R^n$ converging to a point $x^*\in R^n$. Assume that $\forall k$, $x_k \neq x^*$. Given $p \geq 1$, by the *q_p factor*, we mean

$$q_p\{x_k\} = \overline{\lim}_k \frac{\|x_{k+1}-x^*\|}{\|x_k - x^*\|^p}.$$

Observe that the q_p *factor* is norm dependent if $q_p \in (0,\infty)$, while it is norm independent if it is equal to 0 or ∞. We say that the *q-convergence rate* of the sequence is *at least p* if

$$Q_p\{x_k\} < +\infty. \tag{A.1}$$

Moreover, we say that the *q-convergence rate* of the sequence is (*exactly*) p^* where p^* is the supremum of all p such that the q-convergence order is at least p. In the case that $q_1\{x_k\} = +\infty$ we will say that the q-convergence rate is one.

Notice that when we say that the q-convergence rate is p, then p is uniquely determined; however, when we say that the q-convergence rate is at least p, then p is not uniquely determined.

Observe that (A.1) is equivalent to: there exists $c \geq 0$ and k_0 such that

$$\|x_{k+1}-x^*\| \leq c\|x_k - x^*\|^p \quad \forall k \geq k_0.$$

The following convergence notions will be of particular interest to us in the current study.

Again consider a sequence $\{x_k\}\subset R^n$ converging to a point $x^* \in R^n$. Assume that $\forall k$, $x_k \neq x^*$. We say that the convergence of $\{x_k\}$ to x^* is

(i) q-linear if $q_1\{x_k\} < 1$
(ii) q-quadratic if $q_2\{x_k\} < +\infty$
(iii) q-cubic if $q_3\{x_k\} < +\infty$

Observe that a sequence which is q-cubically convergent is also q-quadratically convergent and q-linearly convergent. This slight abuse of terminology will not create problems and is greatly preferred to the alternative of defining these notions so that they are mutually exclusive.

In addition to the q-convergence notions, we have the following r-convergence notions. As before, consider a sequence $\{x_k\} \subset R^n$ converging to a point $x^* \in R^n$. Assume that $\forall k$, $x_k \neq x^*$. We say that the convergence of $\{x_k\}$ to x^* has a particular *r-convergence property*, if there exists a sequence of scalars $\{a_k\}$ converging to zero satisfying

$$\|x_{k+1} - x^*\| \leq a_k \qquad \forall k$$

and the convergence of $\{a_k\}$ to zero has the corresponding q-convergence property.

It follows that q-convergence rate is a stronger notion of convergence than is r-convergence rate. The primary distinction between the two is essentially one of consistent behavior of successive sequence elements. We see that q-convergence guarantees the particular behavior at each element of the sequence, whereas r-convergence guarantees only that the particular property will be demonstrated on average.

The following facts are well-known, moreover their proof are reasonably straightforward

(i) q-linear convergence is norm dependent; all other q and r convergence notions are norm independent.

(ii) The r-order is greater than or equal to the q-order and may be larger.

(iii) If a sequence has a particular q-convergence property, then it has the same r-convergence property.

(iv) If a sequence $\{x_k, y_k\}$ has a particular q-convergence property, then the sequences $\{x_k\}$ and $\{y_k\}$ have the same r-convergence property, but not necessarily the same q-convergence property.

The definitive study of the convergence notion we have described can be found in Chapter 9 of Ortega and Rheinboldt [18]. The qualifiers q and r are due to Ortega and Rheinboldt; they use q to signify quotient factors and r to signify root factors. However, their development of r notions is not what has been presented here. Our development of r-notions has followed Dennis when in 1967 he introduced the notion of pseudo-convergence and it is now known to be equivalent to Ortega and Rheinboldt's notion of r-convergence. The distinction between q and r convergence is quite meaningful and useful and is essentially sacred to workers in the area of computational optimization. However, for not well-understood reasons, computational scientists who are not computational optimizers seem to at best be only tangentially aware of the distinction.

A.2. Algorithm convergence. For the sake of directness and simplicity we will pose and study our various Newton's method formulations in the general framework of functional iteration. It is well-known that for Newton's method and related algorithms such a general approach requires stronger assumptions on differentiation, than does the standard Newton's method analysis. However, this is not an issue in our eigenvalue problem study, since all our functions are infinitely smooth. The general framework of functional iteration offers us flexibility and has a certain degree of pedagogical convenience and value.

Consider $S\colon R^n \to R^n$. We say that x^* is a fixed point of S if $x^* = S(x^*)$. Fixed point problems motivate the iterative procedure

$$x_{k+1} = S(x_k) \quad k = 0, 1, 2, \ldots \tag{A.2}$$

The functional iteration procedure is said to be locally convergent to a fixed point x^* if the sequence generated according to (A.2) converges to x^* for all x_0 in an open set $D(x^*)$ containing x^*. It is said to be linearly, quadratically, or cubically convergent to x^* if all sequences which converge to x^* converge linearly, quadratically, or cubically respectively to x^* as defined in Section 2.1. The following convergence theorem will play a basic role in our analysis.

Theorem A.1. *Consider $S : R^n \to R$ with fixed point x^*. Assume that S has two continuous derivatives in D, a neighborhood of x^*, and*

(i) $S'(x^*) = 0$.

Then functional iteration (A.1) is locally q-quadratically to x^*. If S has three continuous derivations in D and in addition to

(i)' $S'(x^*) = 0$, *we also have*
(ii)" $\|S''(x^*)(x - x^*, x - x^*)\| \leq K\|x - x^*\|^3$

for some constant K and for all $x \in D$, then functional iteration (A.1) is locally q-cubically convergent to x^*.

Proof. The proofs of both parts are direct application of Taylor's theorem with remainder. For the first part we write

$$\begin{aligned}\|x_+ - x^*\| &= \|S(x) - S(x^*)\| = \|S'(x^*)(x - x^*) + \frac{1}{2}S''(x + \theta(x - x^*))(x - x^*, x - x^*)\| \\ &\leq \frac{1}{2}\|S''(x^* + \theta(x - x^*))\|\|x - x^*\|^2, \quad \text{for some } \theta \in (0, 1) \\ &\leq M\|x - x^*\|^2.\end{aligned}$$

The continuity of the second derivative allows us to choose such a constant M.
For the second part we obtain in an analogous fashion

$$\|x_+ - x^*\| \leq (K + M)\|x - x^*\|^3. \qquad \square$$

If we consider Newton's method for the nonlinear equation $F(x) = 0$, then we see that we can write it as functional iteration for the fixed point problem $x^* = S(x^*)$ where

$$S(x) = x - F'(x)^{-1}F(x).$$

For the nonlinear equation $F(x) = 0$, the standard Newton's method theory assumptions for local and q-quadratic convergence of Newton's method are the following:

B1. There exists $x^* \in R^n$ such that $F(x^*) = 0$.
B2. The Jacobian operator F' exists and is Lipschitz continuous in an open neighborhood of x^*.

B3. The Jacobian matrix $F'(x^*)$ is nonsingular.

For a convergence proof and further detail see page 90 of Dennis and Schnabel [6].

Our functional iteration approach to the local q-quadratic convergence of Newton's method uses B1 and B3, and replaces B2 with a slightly stronger smoothness requirement. In what follows we shall assume that we always have as many derivatives as the theory demands. As previously stated this is not an issue in our eigenvalue application.

We now show that appropriate smoothness, existence of a solution, and invertibility of the Jacobian matrix at the solution are sufficient conditions for the local q-quadratic convergence of Newton's method, and if the solution also has 2-norm one for the local q-quadratic convergence of the 2-norm normalized Newton's method.

Our point here is that, while it may not be obvious, the 2-norm normalization does not hurt the rapid convergence of Newton's method under the standard assumptions provided, of course, that the solution has 2-norm equal to one.

THEOREM A.2. *Consider $F\colon R^n \to R^n$. Assume*

(i) There exists x^* such that $F(x^*) = 0$.
(ii) F is sufficiently smooth in a neighborhood of x^*.
(iii) The Jacobian matrix $F'(x^*)$ is nonsingular.

Let

$$N(x) = x - F'(x)^{-1}F(x). \tag{A.3}$$

Then $N'(x^*) = 0$ and Newton's method is locally q-quadratically convergent to x^*. If in addition we have

(iv) $\|x^*\| = 1$,

and we let

$$S(x) = \frac{N(x)}{\|N(x)\|}, \tag{A.4}$$

then $S'(x^*) = 0$ and the normalized Newton's method is locally q-quadratically convergent to x^*.

Proof. From (A.3) we have

$$F'(x)N(x) = F'(x)(x) - F(x).$$

Implicit differentiation at x^* leads to

$$F'(x^*)N'(x^*) + F''(x^*)(N(x^*) - x^*, \cdot) = 0.$$

Recall that $x^* = N(x^*)$ to obtain

$$F'(x^*)N'(x^*) = 0.$$

It follows by nonsingularity of $F'(x^*)$ that $N'(x^*) = 0$. For the second part write (A.4) as

$$\|N(x)\|S(x) = N(x). \tag{A.5}$$

Now observe that if $P(x) = \|x\|_2$, then $P(x)^2 = x^T x$; hence

$$P'(x)(\eta) = \frac{x^T \eta}{\|x\|_2}. \tag{A.6}$$

Implicit differentiation at x^* of (A.5) using (A.6) leads to

$$\|N(x^*)\| S'(x^*) = (I - S(x^*)^T S(x^*) N'(x^*).$$

It follows that $S'(x^*) = 0$, since $N'(x^*) = 0$ and $\|N(x^*)\| = 1$. Recall that $N(x^*) = x^*$ and $\|x^*\| = 1$. □