1967

Power series eigenvalue analysis

Ronald Dean Riess

Iowa State University

Follow this and additional works at: https://lib.dr.iastate.edu/rtd

Part of the Mathematics Commons

Recommended Citation

Riess, Ronald Dean, "Power series eigenvalue analysis" (1967). Retrospective Theses and Dissertations. 3423.
https://lib.dr.iastate.edu/rtd/3423

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.
This dissertation has been microfilmed exactly as received 68-2855

RIESS, Ronald Dean, 1940–
POWER SERIES EIGENVALUE ANALYSIS.

Iowa State University, Ph.D., 1967
Mathematics

University Microfilms, Inc., Ann Arbor, Michigan
POWER SERIES EIGENVALUE ANALYSIS

by

Ronald Dean Riess

A Dissertation Submitted to the
Graduate Faculty in Partial Fulfillment of
The Requirements for the Degree of
DOCTOR OF PHILOSOPHY

Major Subject: Mathematics

Approved:

Signature was redacted for privacy.

In Charge of Major Work

Signature was redacted for privacy.

Head of Major Department

Signature was redacted for privacy.

Dean of Graduate College

Iowa State University
Of Science and Technology
Ames, Iowa

1967
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTRODUCTION AND LITERATURE REVIEW</td>
<td>1</td>
</tr>
<tr>
<td>THE GENERAL PROCEDURE</td>
<td>7</td>
</tr>
<tr>
<td>A GENERAL ANALYSIS OF ERROR MAGNITUDE</td>
<td>10</td>
</tr>
<tr>
<td>THE EXPONENTIAL METHOD</td>
<td>26</td>
</tr>
<tr>
<td>OTHER POWER SERIES TECHNIQUES</td>
<td>54</td>
</tr>
<tr>
<td>CONCLUSION</td>
<td>63</td>
</tr>
<tr>
<td>BIBLIOGRAPHY</td>
<td>66</td>
</tr>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>67</td>
</tr>
</tbody>
</table>
INTRODUCTION AND LITERATURE REVIEW

A principal rule in numerical techniques is to make optimal use of all available data. Let us consider the algebraic eigenvalue problem for finite-dimensional matrices in this context. The traditional power method gives a good illustration of this idea. Given an arbitrary nxn matrix A whose eigenvectors \( \mathbf{u}_1, \ldots, \mathbf{u}_n \); span the n dimensional complex valued vector space \( \mathbb{C}^n \), and also given an arbitrary initial vector \( \mathbf{b}_0 \), the power method builds the vector sequence \( \mathbf{b}_k, k=1, \ldots, N \), where \( \mathbf{b}_k = A \mathbf{b}_{k-1} \) for \( k=1, \ldots, N \). If \( \lambda_1 \) are the eigenvalues of A, and \( \lambda_1 \) is dominant in absolute value, then the vector \( \lambda_1 \mathbf{u}_1 \), will dominate each \( \mathbf{b}_k \) and \( \lambda_1 \) can be approximated by the ratio of the i\(^{th}\) component, \( b_{ki} \), of \( \mathbf{b}_k \) and the i\(^{th}\) component, \( b_{k-1,i} \), of \( \mathbf{b}_{k-1} \) for \( k \) sufficiently large.

This procedure can be extended to two or more dominant roots with equal modulus, and this is done in most fundamental texts in numerical analysis such as Faddeev and Faddeeva (1963) or Bodewig (1959).

Now in problems arising in mathematical physics or engineering, it is often necessary to find only a few of the first eigenvalues of greatest modulus, and in this case the power method can be used to good advantage. But if the problem is to find all of the eigenvalues, the information on the remaining eigenvalues gained through building the
vector sequence \{b_k\}^N_{k=1} is not utilized. The power method
can be adapted to find all of the eigenvalues through a de-
flation technique such as Hotelling's to eliminate eigen-
values previously found, but this entails a great deal of
computation and essentially requires repeating the entire
process at each stage on a new matrix.

Going further in this direction, Lanczos (1955) intro-
duces a method which he calls "spectrographic analysis."
This method is applicable to matrices with distinct real
eigenvalues of modulus less than one. Using the same
notation as above, this method begins by constructing the
vector sequence: \( \hat{b}_1 = \lambda_0 \) and \( \hat{b}_k = 2A\hat{b}_{k-1} - \hat{b}_{k-2} \) (for
\( k=2,...,N \)). Thus \( \hat{b}_k = T_k(\lambda_1)\hat{u}_1 + ... + T_k(\lambda_n)\hat{u}_n \), where the
\( T_k \)'s represent the \( k \)th Chebyshev polynomials. If we let
\( \lambda_i = \cos \theta_i \), \( 0 \leq \theta_i < \pi \), then since \( T_k(t) = \cos(k \cos^{-1}t) \),
\( \hat{b}_k = (\cos k\theta_1)\hat{u}_1 + ... + (\cos k\theta_n)\hat{u}_n \).

Now the vector sequence \{\( \hat{b} \_k \}^N_{k=0} \) is formed where
\( \hat{y}_m = \hat{b}_0 + \hat{b}_1 \cos \frac{m\pi}{N} + ... + \hat{b}_{n-1} \cos \frac{(n-1)m\pi}{N} + \hat{b}_n \cos \frac{Nm\pi}{N} \)
for \( m=0,...,N \).

One next analyzes the behavior of the sequence of the
\( i \)th component of each \( \hat{y}_m \) to derive the \( \theta_i \)'s. This is done
in a case by case manner in Lanczos (1955) and Faddeev and
Faddeeva (1963). Since the constants of the linear combina-
tion of eigenvectors for the \( i \)th component of \( \hat{b}_0 \) are also
obtained, consideration of the other components yield the
set of eigenvectors.

This is a step further in using all computed information, but the computation of the eigenvalues is still done from using one component of the computed vector sequence plus the fact that it is somewhat limited in its scope of application.

The author develops in this paper a general technique which uses all of this computed information in obtaining the eigenvalues of an nxn matrix A which has a complete set of n eigenvectors. The technique entails choosing a function G(x) such that the vector function \( \hat{H}(x) = G(\lambda_1 x)u_1 + \ldots + G(\lambda_n x)u_n \) can be constructed and such that by a given number of evaluations of \( \hat{H}(x) \) the individual \( \lambda_i \)'s, and possibly the \( u_i \)'s also, can be recovered. This recovery scheme plus the character of the function \( G(x) \) in constructing \( \hat{H}(x) \) will determine the applicability of the method. One sees also that for each \( G(x) \) and each recovery scheme there is a different technique to apply to his matrix. Thus one may use different approaches to different classes of matrices. The author will examine three schemes which exemplify different modes of attack, all within this general structure.

The first obstacle to be encountered, however, is the amount of computational error in general encountered through such a process. A natural way of constructing \( \hat{H}(x) \) is through a power series technique, and this will be the procedure to which the error analysis will be applied. J. H.
Wilkinson has published several papers on round-off and truncation error in different computational procedures, and has consolidated them, along with papers by other authors, in his book *Rounding Errors in Algebraic Processes*. His error analysis of individual operations will be used to construct an error analysis of the method in general. For round-off error, a forward error analysis, which gives a comparison between computed values and actual values at each step, will be given first; and then a backward error analysis is given, which gives the computed value at various stages of the process in terms of the actual value and relative error of the parameters. The backward analysis is completed by giving bounds for the effect of the perturbation of parameters on the actual values of $\hat{H}(x)$. In doing this the author shows how soon the round-off error becomes dominant over the truncation error, but that the total error is not so large that the method is not practical.

The first technique that the author considers is for $G(x) = e^{hx}$ and $\hat{H}(x)$ to be built up by either a Maclaurin series or by iteratively solving a simple system of ordinary differential equations. This technique will be treated carefully to illustrate different modes of attack.

This first technique essentially finds the characteristic polynomial of the matrix $e^{hA}$, where $h$ is an arbitrary positive constant. Now there are several techniques in the
literature which iterate to find the characteristic polynomial of a matrix. Probably the simplest is one which Faddeev and Faddeeva (1963) attribute to A. N. Krylov, and which is generally associated with the power method. In this technique one again forms the vector sequence
\[ \hat{\mathbf{b}}_k = A \hat{\mathbf{b}}_{k-1}; \text{ for } k=1, \ldots, n; \text{ from some arbitrary vector } \hat{\mathbf{b}}_0. \]
Now
\[ \hat{\mathbf{b}}_n = \sum_{i=0}^{n-1} c_i \hat{\mathbf{b}}_i \]
where the \(c_i\)'s are the coefficients of the characteristic polynomial of \(A\) if all of the eigenvalues are distinct.

There are other techniques to find the characteristic polynomial of \(A\), but they all have one common disadvantage, and thus are not in popular use. This disadvantage is that the characteristic polynomial of \(A\) can be ill-conditioned and thus a small perturbation in the coefficients can cause a large perturbation in the roots. There is almost always some error present in the coefficients due to their calculation, and thus the aspect of conditioning plays an important role. In the technique presented here with \(G(x) = e^{+x}\), the author shows that the exponential characteristic polynomial can be made well-conditioned with respect to perturbation in any one coefficient by a proper choice of \(h\). Thus the usual inhibiting factor for methods of this type is removed. This stability will be demonstrated both by numerical example and by application to an example treated extensively by

The last topic covered will be a less extensive treatment of the general method, where in one case $G(x)$ is chosen to have maxima at the reciprocals of the eigenvalues, and in the other case where $G(x)$ is set up to suppress eigenvalues outside of a given interval.
THE GENERAL PROCEDURE

Let us consider an arbitrary nxn matrix A whose eigenvectors \( \hat{u}_1, \ldots, \hat{u}_n \) span the complex-valued n-dimensional vector space \( C^n \). Assume that an initial vector \( \hat{b}_0 \) can be expressed as a non-zero linear combination of eigenvectors of A, and by absorbing the constants of this linear combination into the eigenvectors, we have:

\[
\hat{b}_0 = \hat{u}_1 + \ldots + \hat{u}_n.
\]

We next form the vector sequence \( \{\hat{b}_k\}_{k=1}^N \), where

\[
\hat{b}_1 = A\hat{b}_0 \quad \text{and} \quad \hat{b}_k = A\hat{b}_{k-1}; \quad k=1, \ldots, N;
\]
and \( N \) is a positive integer to be determined below.

We now consider a function \( G(x) \) which possesses a Maclaurin series expansion with a region of convergence containing all points \( x \) such that \( |x|<a \) where \( a>0 \) and we shall only consider points such that \( |x|<a/M \) where \( M \) is some bound on the spectral radius of \( A, \rho(A) \). Let this series be given as:

\[
G(x) = \sum_{k=0}^N a_k x^k = \sum_{k=0}^N a_k x^k + R_N(x).
\]

We choose \( N \) so that the remainder, \( R_N(x) \), of this series is less than some prescribed \( \varepsilon>0 \) for any point \( x: |x|<a/M \).

Now for a given value of \( x \), we multiply \( \hat{b}_k \) by \( x^k \) and \( a_k \). We then have the following series where \( \lambda_1, \ldots, \lambda_n \) denote the eigenvalues of \( A \):
\[ \hat{H}_N(x) = a_0 \hat{b}_0 + a_1 \hat{b}_1 x + \ldots + a_N \hat{b}_N x^N \]
\[ = a_0 (\hat{u}_1 + \ldots + \hat{u}_n) + a_1 (\lambda_1 x \hat{u}_1 + \ldots + \lambda_n x \hat{u}_n) \]
\[ + \ldots + a_N (\lambda_1^N x^N \hat{u}_1 + \ldots + \lambda_n^N x^N \hat{u}_n) \]
\[ = (a_0 + a_1 \lambda_1 x + \ldots + a_N \lambda_1^N x^N) \hat{u}_1 + \ldots \]
\[ + (a_0 + a_1 \lambda_n x + \ldots + a_N \lambda_n^N x^N) \hat{u}_n. \]

Let \( \hat{H}(x) = \lim_{N \to \infty} \hat{H}_N(x) = G(\lambda_1 x) \hat{u}_1 + \ldots + G(\lambda_n x) \hat{u}_n. \)

Then \( \hat{H}_N(x) = G(\lambda_1 x) \hat{u}_1 + \ldots + G(\lambda_n x) \hat{u}_n, \) for \( N \) sufficiently large. We now have an approximation to the vector valued function \( \hat{H}(x) \) whose region of convergence contains all points \( x \) such that \( |\lambda_1 x| < M |x| < a. \)

Our objective is to recover the eigenvalues of \( A \) and also possibly the eigenvectors. To do this we see that we are able to evaluate \( \hat{H}(x) \) at as many points within its radius of convergence as we desire. Thus we can see the need for a judicious choice for \( G(x) \). We shall later discuss three such choices for \( G(x) \) which enable us to efficiently recover the eigenvalues of \( A \), where each illustrates a different possible recovery scheme. We shall also see later that there are different ways of evaluating \( G(x) \) other than by a Mac-laurin series. It is our belief that there are several good eigenvalue methods which are of this category.

As far as speed of computation is concerned, it is noted that the vector sequence need only be computed once. The evaluations for different \( x \)'s are just the results of the multiplication of a scalar times a vector and the subsequent
addition. Thus these different evaluations are themselves relatively fast. The two most dominant features to be considered in choosing $G(x)$, then, are:

1.) The ease of recovering the $\lambda_i$'s from different evaluations of $\hat{H}(x)$ and

2.) The number of iterations $N$ necessary to make the power series remainder for $G(x)$ to be as small as desired.

We see that round-off and truncation error can play a large part in both of these considerations and thus that is the topic we discuss next.
A GENERAL ANALYSIS OF ERROR MAGNITUDE

Let us consider this forward error analysis from the viewpoint of correlating round-off error and truncation error. Let $S = \sup_{k \in K} \sup_{t \in R} G^{(k)}(t)$, where $R$ is the above region of convergence and $K$ is the set of non-negative integers.

The error from the series truncation is

$$R_N(t) = \frac{G^{(N+1)}(\xi)t^{N+1}}{(N+1)!}, \text{ for } |\xi| < |t|.$$ 

Then

$$R_N(t) \leq \frac{S \cdot a}{(N+1)!}.$$ 

Let us consider a machine representation of 16 hexadecimal digits or 64 binary places. Then we choose $N$ such that

$$\frac{S \cdot a}{(N+1)!} < 2^{-64},$$

and the truncated terms are negligible to 64 binary places.

Let us build up the power series for $G(t)$ by the following nested multiplication. Let

$$p(t) = \sum_{k=0}^{N} \frac{G^{(k)}(0)t^k}{k!} \equiv \sum_{k=0}^{N} a_k t^k.$$ 

Let

$$\overline{p}(t) = \sum_{k=0}^{N} \overline{a_k} t^k,$$

where $\overline{a_k} = a_k(1 + e_k)$ is the machine representation for each $a_k$. Let us assume that $m$ is the number of binary places necessary to contain the floating point sign, exponent, and decimal point coding. Then the mantissa of each $a_k$ contains
(64-m) binary places. The rounding can be at most 1/2 of the last binary digit, so
\[ |e_k| \leq 2^{-(64-m+1)}. \]
Hereafter let \( m_1 = 64-m+1 \) and \( m_2 = 64-m \).

Now make the following definitions:
\[ u_N(t) = \bar{a}_N \quad \text{and} \quad u_k(t) = fl(tu_{k+1}(t) + \bar{a}_k); \]
\[ k = N-1, N-2, \ldots, 0; \]
where \( fl \) denotes the floating point representation of a quantity.

Wilkinson (1963) shows that if \( \tau \) is the number of binary places allowed to the mantissas and the operations are done with a double precision accumulator, the following are true:

1.) \( fl(x_1 + x_2) = (x_1 + x_2)(1 + \varepsilon_1) \) and

2.) \( fl(x_1x_2) = x_1x_2(1 + \varepsilon_2), \) where \( |\varepsilon_1| \leq 2^{-\tau}. \)

Thus \( u_k(t) = tu_{k+1}(t)(1 + f_k)(1 + g_k) + \bar{a}_k(1 + g_k) \) where \( |f_k| \leq 2^{-m_2} \) and \( |g_k| \leq 2^{-m_2} \), are the round-off errors due to the indicated multiplication and addition, respectively.

Proceeding recursively from \( N \) to 0 we see that
\[ u_0(t) = \bar{p}(t) = \sum_{k=0}^{N} \bar{a}_k t^k, \]
the machine representation of \( G(t) \). In doing these products and sums we build up products of these relative error terms, and finally get:
\[ u_0(t) = a_0(1 + E_0) + a_1(1 + E_1)t + \ldots + a_N(1 + E_N)t^N \]
where \((1 - 2^{-m_1})^{2k+2} \leq (1 + E_k) \leq (1 + 2^{-m_1})^{2k+2}\).

By using binomial expansions for these two bounds, and regrouping all of the terms except the first and forming another power of a binomial, we can show:

\[(1 + 2^{-m_1})^{2k+2} < 1 + (2k + 2)2^{-m_1}(1 + 2^{-m_1})^{2k+1}\]

and \[(1 - 2^{-m_1})^{2k+2} > 1 - (2k + 2)2^{-m_1}(1 + 2^{-m_1})^{2k+1} .\]

Then \(|E_k| < (2k + 2)2^{-m_1}(1 + 2^{-m_1})^{2k+1} .\)

Now choose \(d\) such that \(d\) is the minimum bound such that \((1 + 2^{-m_1})^{2k+1} \leq 1 + d\) for \(k = 1, \ldots, N;\)

\[i.e., \ d = (1 - 2^{-m_1})^{2N+1} .\]

Thus we get the more workable inequality

\[|E_k| < (1 + d)(k + 1)2^{-m_2} .\]

Then \(|u_n(t) - p(t)| = |E_0 a_0 + E_1 a_1 t + \ldots + E_N a_N t^N| \leq |E_0^2 + E_1^2 + \ldots + E_N^2|^{1/2} |a_0^2 + a_1^2 t^2 + \ldots + a_N^2 t^2N|^{1/2}\]

by Schwartz's inequality.

Now \(|a_k| \leq \frac{s}{k!}\)

so \(|a_k^{2k}| \leq \frac{s^2}{(k!)^2} \sup \ |t|^{2k} = \frac{s^2 a^{2k}}{(k!)^2} .\)

If \(k_1 = [a]\), then the largest bound for \(|a_k^{2k}|\) occurs when \(k = k_1\), and thus \(|a_k^{2k}| \leq \frac{s^2 a^{2k_1}}{(k_1 !)} .\)

Up to this point this analysis has been kept general, but let us now put some numerical bounds on these parameters to see how large this round-off error bound is. If we let
m=6, S=6 and a=4, then for N=32 we have the truncation error less than $2^{-65}$. For N=32, d can be shown to be less than .01. Then we get

$$|a_k2k| \lesssim \frac{36 \cdot 4^8}{(24)^2} = 4^6.$$  

If $E = \max_{1 \leq i \leq N} \{ |E_i| \}$, then $E \leq (1.01)(33)^{2^{-60}}$.

Then

$$|u_o(t) - p(t)| \leq \left| 33E^2 \right|^{1/2} \left| (33)^{4^6} \right|^{1/2}$$

$$= 33E^{4^3} \leq (33)^2(1.01)^2 \cdot 2^{-60}$$

$$\leq 2^{-43}.$$  

Now the $k!$ dominates $a_k t^k$ as $k$ becomes large, so that we see that $\sum a_k t^k$ is a poor bound for all $a_k t^k$.

Let us compare this bound for $|u_o(t) - p(t)|$ with one we derive using Abel's summation identity (Taylor, 1955), i.e., if $\{b_k\}_{k=0}^{\infty}$ and $\{c_k\}_{k=0}^{\infty}$ are arbitrary sequences and

$$S_n = b_0 + \ldots + b_n,$$

then

$$b_0 c_0 + b_1 c_1 + \ldots + b_n c_n$$

$$= S_0(c_0 - c_1) + S_1(c_1 - c_2) + \ldots + S_n - 1(c_{n-1} - c_n) + S_n c_n.$$  

Let $E_k = c_k$ and $a_k t^k = b_k$.

Now $G(t) = \sum_{k=0}^{j} a_k t^k + R_j(t)$; for $j=0,1,2,\ldots$;

where

$$R_j(t) = \frac{G^{j+1}(E_j)t^{j+1}}{(j+1)!}, 0 \leq |E_j| \leq |t|.$$  

Then under our assumptions
\[ |R_j(t)| \leq \frac{S \cdot a^{j+1}}{(j+1)!} \equiv R_j \]

and thus
\[ \left| \sum_{k=0}^{j} a_j t^j \right| \leq |G(t)| + |R_{j+1}(t)| \leq S + S \cdot a^{j+1} \]

Also \( |E_{k-1} - E_k| \leq |E_{k-1}| + |E_k| \leq (1+d)k2^{-m_2} \)
+ \( (1+d)(k+1)2^{-m_2} = (1+d)(2k+1)2^{-m_2} \).

Now using Abel's equality we get:
\[ |u_0(t) - p(t)| = \left| E_0 a_0 + E_1 a_1 t + \ldots + E_N a_N t^N \right| \]
\[ = \left| a_0 (E_0 - E_1) + a_1 t (E_1 - E_2) + \ldots \right| + \left| a_0 + a_1 t + \ldots + a_{N-1} t^{N-1} (E_{N-1} - E_N) \right| \]
\[ + \left| a_0 + a_1 t + \ldots + a_N t^N E_N \right| \]
\[ = \left| (G(t) - R_1) (E_0 - E_1) + (G(t) - R_2) (E_1 - E_2) + \ldots \right| + \left| (G(t) - R_N) (E_{N-1} - E_N) + (G(t) - R_N+1) E_N \right| \]
\[ = \left| G(t) E_0 - (R_1 (E_0 - E_1) + R_2 (E_1 - E_2) + \ldots \right| + \left| R_N (E_{N-1} - E_N) + R_{N+1} E_N \right| \]
\[ \leq |G(t)| \left| E_0 \right| + |R_1| \left| E_0 - E_1 \right| + |R_2| \left| E_1 - E_2 \right| + \ldots \]
\[ + \left| E_{N-1} - E_N \right| + \left| E_N \right| \]
\[ \leq S \left| E_0 \right| + S \left( a \left| E_0 - E_1 \right| + \frac{a^2}{2} \left| E_1 - E_2 \right| + \ldots \right| \]
\[ + \frac{a^N}{N!} \left| E_{N-1} - E_N \right| + \frac{a^{N+1}}{(N+1)!} \left| E_N \right| \]
\[ \leq S \left| E_0 \right| + S (1+d)2^{-m_2} \sum_{k=0}^{N} \frac{(2k+1) a^{k+1}}{(k+1)!} \]

Now \[ \sum_{k=0}^{N} \frac{(2k+1) a^{k+1}}{(k+1)!} \leq \sum_{k=0}^{\infty} \frac{(2k+1) a^{k+1}}{(k+1)!} \]
\[
= \sum_{k=0}^{\infty} \left( \frac{(2k+2)a^{k+1}}{(k+1)!} - \frac{a^{k+1}}{(k+1)!} \right)
\]

\[
= 2\sum_{k=0}^{\infty} \frac{(k+1)a^{k+1}}{(k+1)!} - \sum_{k=0}^{\infty} \frac{a^{k+1}}{(k+1)!}
\]

\[
= 2a\sum_{k=0}^{\infty} \frac{a^k}{k!} - \sum_{k=1}^{\infty} \frac{a^k}{k!}
\]

\[
= 2a \cdot e^a - (e^a - 1) = (2a-1)e^a + 1.
\]

Again we let \(m=6, S=6\) and \(a=4\) to see how big this error bound is in a special case.

Then

\[
|u_0(t) - p(t)| \leq 6(1.01)2^{-60} + 6(1.01)2^{-60} \cdot e^4
\]

\[
< (6.06)2^{-60} + (6.06)2^{-60}(386)
\]

\[
< 2^{-57} + 2^{-48}.
\]

Thus the round-off error is less than \(2^{-47}\) or correct to approximately 14 decimal places. For a more specific \(G(t)\), a much tighter bound can be found. But, in any case, the truncation error bound of \(2^{-65}\) is much lower than the round-off error bound, indicating that there is probably no advantage in iterating further and achieving a lower truncation error.

Now we have a bound on the error accumulated through building a Maclaurin series for a function. Since this seems to be a reasonable bound, we now go on to consider an
error bound on the development of \( \hat{H}(x) \) for any \( x \). In doing this we will give a backward error analysis, considering the actual values in terms of the computed values and relative errors at various stages of the process. These errors will be in vector form and the bounds will be given in terms of the norms of these vectors. A comparison of the computed value and actual value will be given only at the end to conclude this analysis.

First of all we see from Wilkinson (1963) that the round-off error from forming an inner product can be approached in the following manner.

Define \( s_r \) and \( t_r \) recursively such that:

\[
\begin{align*}
t_r &= f_l(a_r b_r), \\
s_1 &= t_1, \quad \text{and} \quad s_r &= f_l(s_{r-1} + t_r),
\end{align*}
\]

for \( r=1, \ldots, N \).

Thus we see

\[
\begin{align*}
t_r &= a_r b_r (1 + \xi_r) \quad \text{and} \quad s_r &= (s_{r-1} + t_r)(1 + \eta_r),
\end{align*}
\]

where \( |\xi_r| \leq 2^{-r} \), \( |\eta_r| \leq 2^{-r} \), \( r \) equals the number of binary places at our disposal, and \( f_l(x) \) represents the floating point representation of \( x \).

Then

\[
s_N = a_1 b_1 (1 + \xi_1) + a_2 b_2 (1 + \xi_2) + \ldots + a_N b_N (1 + \xi_N),
\]

where

\[
1 + \xi_1 = (1 + \xi_1)(1 + \eta_2) \ldots (1 + \eta_N)
\]
and \[ 1 + \xi_r = (1 + \xi_r)(1 + \eta_r) \cdots (1 + \eta_N) \]
for \( r = 2, \ldots, N \).
Then
\[ (1 - 2^{-r})^N \leq (1 + \xi_1) \leq (1 + 2^{-r})^N \]
and
\[ (1 - 2^{-r})^{N-r+2} \leq (1 + \xi_r) \leq (1 + 2^{-r})^{N-r+2} \]
for \( r = 2, \ldots, N \).

By a similar procedure as before using the binomial expansion of \((1 + 2^{-r})^r\), and restricting \( r \) such that \( r \cdot 2^{-r} < 0.1 \), it can be shown that
\[ (1 + 2^{-r})^r < 1 + (1.05) r \cdot 2^{-r} \]
and
\[ (1 - 2^{-r})^r > 1 - (1.05) r \cdot 2^{-r} . \]

Using Wilkinson's notation we define \( t_1 \) such that
\[ 2^{-t_1} = (1.05) 2^{-r} . \]

Now inequalities of the form
\[ (1 - 2^{-r})^r \leq 1 + \xi \leq (1 + 2^{-r})^r \]
can be expressed in the simpler form
\[ |\xi| < r \cdot 2^{-t_1} . \]

Now let us consider the round-off error in the computation of \( A\hat{x} \), where \( A \) is an \( nxn \) matrix and \( \hat{x} \) is an \( nx1 \) vector.

Now we let
\[ y_i = f^1(a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n) = a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n + \epsilon_i, \]
where, from the above work,
\[ |\epsilon_i| \leq 2^{-t_1} \left( n |a_{i1}| |x_1| + n |a_{i2}| |x_2| + \ldots + 2 |a_{in}| |x_n| \right) . \]
Then \[ \hat{y} = A\hat{x} + \hat{e}. \]

If \[ |\hat{e}|^T = (|e_1|, |e_2|, \ldots, |e_n|), \quad |A| = (|a_{ij}|), \]
\[ |\hat{x}|^T = (|x_1|, |x_2|, \ldots, |x_n|), \quad \text{and} \quad D = \begin{bmatrix} n & 0 & \ldots & 0 \\ 0 & n & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \ldots & \ldots & 0 \end{bmatrix}, \]
then \[ 0 \leq |e_1| \leq 2^{-t_1}(|A|D|\hat{x}|)^1. \]

Then
\[
\|\hat{e}\| \leq \|\hat{e}\| \leq 2^{-t_1}\|AD\hat{x}\| = 2^{-t_1}\|A\|\cdot\|D\|\cdot\|\hat{x}\| = 2^{-t_1}\|A\|\cdot\|\hat{x}\|
\]
(Note: All norms will be spectral norms unless otherwise specified.)

The Euclidean norm,
\[
\|A\|_E = \left(\sum_{i=1}^{n} \sum_{j=1}^{n} |a_{ij}|^2 \right)^{1/2},
\]
is usually easier to compute and can be substituted into the above inequalities through the relationship
\[
\|A\| \leq \|A\|_E \leq \sqrt{n}\|A\|.
\]

Now we consider the round-off error accrued in constructing the vector set \( \{\hat{b}_k\}_{k=1}^N \). Now \( \hat{b}_0 \) is an arbitrary vector so we will assume that it is chosen so that it can be exactly represented on the machine. Also let \( \|\hat{b}_0\| = m \).

Now \( f1(\hat{b}_1) = f1(A\hat{b}_0) = A\hat{b}_0 + \hat{e}_1 \).

Let \( \hat{x}_1 = \hat{b}_0 \), then \( \|\hat{x}_1\| = m \).
Then by the preceding results
\[ \| \mathbf{c}_1 \| \leq 2^{-t_1} \| \mathbf{A} \| m. \]
Continuing, we have
\[ f_1(\mathbf{b}_2) = f_1(\mathbf{A}(\mathbf{A}\mathbf{b}_0 + \mathbf{e}_1)) = \mathbf{b}_1 + \mathbf{A}\hat{\mathbf{e}}_1 + \mathbf{e}_2. \]
Let \( \hat{\mathbf{x}}_2 = \mathbf{A}\hat{\mathbf{e}}_0 + \mathbf{e}_1 \), then
\[ \| \hat{\mathbf{x}}_2 \| \leq \| \mathbf{A} \| \| \mathbf{x}_1 \| + \| \mathbf{e}_1 \| \\
\leq \| \mathbf{A} \| m + 2^{-t_1} \| \mathbf{A} \| m \\
= \| \mathbf{A} \| m(1 + n2^{-t_1}). \]
Then
\[ \| \mathbf{e}_2 \| \leq 2^{-t_1} \| \mathbf{A} \| n \| \hat{\mathbf{x}}_2 \| \\
\leq 2^{-t_1} \| \mathbf{A} \| 2mn(1+n2^{-t_1}). \]
Continuing further,
\[ f_1(\mathbf{b}_3) = f_1(\mathbf{A}(\mathbf{A}\hat{\mathbf{x}}_2 + \mathbf{e}_2)) = \mathbf{A}^2\hat{\mathbf{x}}_2 + \mathbf{A}\hat{\mathbf{e}}_2 + \mathbf{e}_3. \]
Let \( \hat{\mathbf{x}}_3 = \mathbf{A}\hat{\mathbf{x}}_2 + \mathbf{e}_2 \), then
\[ \| \hat{\mathbf{x}}_3 \| \leq \| \mathbf{A} \| \| \hat{\mathbf{x}}_2 \| + \| \mathbf{e}_2 \| \\
\leq \| \mathbf{A} \| 2m(1+n2^{-t_1}) + 2^{-t_1} \| \mathbf{A} \| 2mn(1+n2^{-t_1}) \\
= \| \mathbf{A} \| 2m(1+n2^{-t_1})(1+n2^{-t_1}) \\
= \| \mathbf{A} \| 2m(1+n2^{-t_1})^2. \]
Then
\[ \| \mathbf{e}_3 \| \leq 2^{-t_1} \| \mathbf{A} \| n \| \hat{\mathbf{x}}_3 \| \\
\leq \| \mathbf{A} \| 3mn2^{-t_1}(1+n2^{-t_1})^2. \]
Assume
\[ \| \hat{\mathbf{x}}_{k-1} \| \leq \| \mathbf{A} \| ^{k-2} m(1+n2^{-t_1})^{k-2}. \]
Then
\[ \| \mathbf{e}_{k-1} \| \leq 2^{-t_1} \| \mathbf{A} \| 2m(1+n2^{-t_1})^{k-2} \]
\[ \| \mathbf{e}_{k-1} \| \leq 2^{-t_1} \| \mathbf{A} \| \| \hat{\mathbf{x}}_{k-1} \| \\
\leq 2^{-t_1} \| \mathbf{A} \| ^{k-1} n m(1+n2^{-t_1})^{k-2}. \]
Now \( f^l(\hat{e}_k) = f^l(A(A^T\hat{e}_{k-1} + e_{k-1})) = f^l(A\hat{e}_k) \)
\[= A\hat{e}_k + \hat{e}_k.\]

Then
\[
\|\hat{e}_k\| \leq \|A\| \|\hat{e}_{k-1}\| + \|\hat{e}_{k-1}\|
\leq \|A\|^{k-1}m(1+n2^{-t1})^{k-2} + 2^{-t1}\|A\|^{k-1}nm(1+n2^{-t1})
= \|A\|^{k-1}m(1+n2^{-t1})^{k-2}(1+n2^{-t1})
= \|A\|^{k-1}m(1+n2^{-t1})^{k-1}.
\]

Thus inductively we see that
\[
\|\hat{e}_r\| \leq \|A\|^{r-1}nm(1+n2^{-t1})^{r-1}, \text{ for } r=1,...,N.
\]
(Note: \( \hat{e}_r = f^l(\hat{e}_{r-1}) \)).

In this analysis let us only consider matrices which are normalized such that their largest eigenvalue has absolute value less than one. By theorem 1.4 in Varga (1963), we have that in this case \( \lim_{r \to \infty} \|A\|^r = 0. \)

Thus the size of the term \( (1+n2^{-t1})^{r-1} \) is counteracted by \( \|A\|^r \). If we must predivide \( A \) by a constant \( k \) to achieve this property (such as the maximum absolute row or column sum), Wilkinson (1963) shows that the round-off error induced by this step is bounded by \( \|k\|2^{-T}n^{1/2}\|A\| \). Since \( T \) is generally large, \( 2^{-T} \) will usually dominate this expression. In this case \( A \) must be very ill-conditioned, with respect to its eigenvalues, before this division by \( k \) will significantly alter the eigenvalues of \( A \), which in turn equal \( k \) times the eigenvalues of \( A \div k \). In most compilers,
however, this round-off can be completely avoided by dividing by an integral power of the base of the system. For example, the IBM 360 model 50 has a hexadecimal base. Division by 16 in floating point in this system is accomplished simply by subtracting 1 from the floating point exponent. The mantissa is completely untouched, and thus no round-off occurs. But even if it is somehow impractical to divide by a power of the base of the system, the author advocates the above normalizing predivision. Two major reasons are

1.) \( \lim_{r \to \infty} \|A \div k\|^r = 0 \) and

2.) it eliminates the necessity of normalizing each \( \hat{b}_k \) and remultiplying by its length later in computing \( \hat{H}(x) \).

Each of these two factors do much to keep the round-off error small.

Now we have a bound on the round-off error in forming the vector sequence \( \{\hat{b}_k\}_{k=0}^{N} \) given in vector norm form, and a bound on the round-off error in forming a Maclaurin series given in scalar form. In this eigenvalue procedure we must combine the two to get a bound on the total round-off error in forming the vector series:

\[
\hat{H}(t) = G(\lambda_1 t)\hat{u}_1 + \ldots + G(\lambda_n t)\hat{u}_n
\]

\[
= \sum_{k=0}^{N} a_k \hat{b}_k t^k.
\]
We can either look at the error component-wise for each $(\hat{H}(t))^\bot$, or consider in vector norm form for the whole vector $\hat{H}(t)$ at once. We elect to do the latter, since reducing the error in vector form involves the relationship that if $\|\hat{e}_r\| < \varepsilon$ then the best component-wise bound attainable is to say that the modulus of the maximum component of $\hat{e}_r$ is less than or equal to $\varepsilon$. But since $\varepsilon$ depends on all components of $\hat{e}_r$, there is no way to separate the round-off errors due to each component. Thus we will approach the round-off error in this problem from a vector viewpoint.

As stated earlier, $\hat{b}_0$ is chosen such that $fl(\hat{b}_0) = \hat{b}_0$. Then using all notations as above,
\[
\begin{align*}
\hat{x}_1 &= fl(\hat{b}_0) = \hat{b}_0 \\
\hat{x}_2 &= fl(\hat{b}_1) = A\hat{x}_1 + \hat{c}_1 = \hat{b}_1 + \hat{c}_1 \\
\hat{x}_3 &= fl(\hat{b}_2) = A\hat{x}_2 + \hat{c}_2 = \hat{b}_2 + A\hat{c}_1 + \hat{c}_2 \\
&\vdots \\
\hat{x}_{r+1} &= fl(\hat{b}_r) = \hat{b}_r + A^{r-1}\hat{c}_1 + \cdots + A\hat{c}_{r-1} + \hat{c}_r,
\end{align*}
\]
for $r=1,2,\ldots,N$.

Now we want to form the summation
\[
\hat{H}(t) = \sum_{k=0}^N a_k t^k \hat{b}_k.
\]

We have from Wilkinson (1963) that $fl(x_1x_2) = x_1x_2(1+f)$ where
\[
(1-2^{-f}) \leq (1+f) \leq (1+2^{-T}).
\]
Thus $fl(a_k t^k(\hat{b}_k)_i) = a_k t^k(\hat{b}_k)_i (1+f(\hat{b}_k)_i)$; for $i=1,\ldots,n$;
and where \((1-2^{-r})^{k+1} \leq (1+f_{kl}) \leq (1+2^{-r})^{k+1}\).

Then

\[
f_1(a_k t^k b_k) = \begin{pmatrix} a_k t^k (b^r_k) (1+f_{kl}) \\ \vdots \\ a_k t^k (b^r_{kn}) (1+f_{kn}) \end{pmatrix} = a_k t^k (I+F_k) b_k,
\]

where \(F_k = \begin{pmatrix} f_{kl} & 0 & \cdots & 0 \\ 0 & f_{k2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & f_{kn} \end{pmatrix}\); and since \(|f_{kl}| \leq (k+1)2^{-t_1}\),

then

\[
\|F_k\| \leq (k+1)2^{-t_1}
\]

and

\[
\|I + F_k\| \leq 1 + (k+1)2^{-t_1}.
\]

Now, since

\[
X_{k+1} = b_k + A^k e_1 + \cdots + A e_{k-1} + e_k,
\]

we let

\[
\hat{X}_k = a_k t^k b_k - f_1(a_k t^k b_k) = a_k t^k b_k - a_k t^k (I+F_k) b_k = -a_k t^k F_k b_k
\]

\[
= -a_k t^k F_k (X_{k+1} - \sum_{j=1}^{k} A e_j).
\]

Then

\[
\|\hat{X}_k\| \leq |a_k| |t^k| (k+1)2^{-t_1} (\|A\|^{k-1} m (1+n2^{-t_1})^k
\]

\[
+ \sum_{j=1}^{k} \|A\|^{k-1} m n2^{-t_1} (1+n2^{-t_1})^{j-1})
\]

\[
\leq |a_k| |t^k| \|A\|^{k+1} 2^{-t_1} m \left[(1+n2^{-t_1})^k
\right]
\]
Lastly we include the error due to the summation of the above terms.

Let \( \hat{\gamma}_k = f_l(a_k t^k b_k) \),

then \( a_k t^k b_k = \hat{\gamma}_k + \hat{\varepsilon}_k \).

Again \( \hat{b}_o \) is arbitrary, and this time we make the further restriction that it be chosen so that \( a_o \hat{b}_o \) can be represented exactly.

Let \( \hat{\gamma}_o = f_l(a_o \hat{b}_o) = a_o \hat{b}_o = \hat{\gamma}_o \)

and \( \hat{\gamma}_r = f_l(\hat{\gamma}_{r-1} + \hat{\gamma}_r) = (I + \hat{\gamma}_r)(\hat{\gamma}_{r-1} + \hat{\gamma}_r) \),

where \( \hat{\gamma}_r \) is, as others above, a diagonal matrix such that

\[ \| \hat{\gamma}_r \| \leq 2^{-r} \text{ and } \| I + \hat{\gamma}_r \| \leq (1+2^{-r}) \].

Then \( \hat{\gamma}_N = f_l(\sum_{k=0}^{N} a_k t^k b_k) \).

Recursively we can show that \( \hat{\gamma}_r = (I + \hat{\gamma}_1) \ldots (I + \hat{\gamma}_r) \hat{\gamma}_o + \sum_{k=1}^{r} (I + \hat{\gamma}_k) \ldots (I + \hat{\gamma}_r) \hat{\gamma}_k \),

for \( r = 1, \ldots, N \).

Then \( \hat{\gamma}_N = (I + \eta_1) \hat{\gamma}_o + (I + \eta_1) \hat{\gamma}_1 + \ldots + (I + \eta_N) \hat{\gamma}_N \),

where \( (I + \eta_r) = (I + \hat{\gamma}_r) \ldots (I + \hat{\gamma}_N) \).

Then \( \| I + \eta_r \| \leq (1+2^{-r})^{N+1-r} \text{ or } \| \eta_r \| \leq (N+1-r)2^{-t_1} \)

Since \( a_k t^k b_k = \hat{\gamma}_k + \hat{\varepsilon}_k \) and \( \hat{\varepsilon}_o = \hat{\varepsilon} \)
then  \[ \hat{V}_n = a_0 (I + \eta_1) \hat{V}_0 + (I + \eta_1)(a_1 t \hat{b}_1 - \hat{\xi}_1) \]
\[ + (I + \eta_1)(a_2 t^2 \hat{b}_2 - \hat{\xi}_2) + \ldots \]
\[ + (I + \eta_1)(a_N t^N \hat{b}_N - \hat{\xi}_N) \]
\[ = \sum_{k=0}^{N} a_k t^k \hat{b}_k + a_0 \eta_1 \hat{b}_0 + \sum_{k=1}^{N} a_k t^k \eta_k \hat{b}_k - \sum_{k=1}^{N} (I + \eta_k) \hat{\xi}_k. \]

Therefore the expression for the accumulated round-off error

is  \[ \hat{\xi}(t) = \sum_{k=0}^{N} a_k t^k \hat{b}_k - R \left( \sum_{k=0}^{N} a_k t^k \hat{b}_k \right) \]
\[ = -a_0 \eta_1 \hat{b}_0 - \sum_{k=1}^{N} a_k t^k \eta_k \hat{b}_k + \sum_{k=1}^{N} (I + \eta_k) \hat{\xi}_k. \]

The bound on the norm of this vector is

\[ \| \hat{\xi}(t) \| \leq |a_0| N^{2-t_1} - \sum_{k=1}^{N} |a_k| |t^k| (N-k+1)2^{-t_1} \| A \|^{k-m} \]
\[ + \sum_{k=1}^{N} \left[ (1+2^{-t_1})^{N-k} |a_k| |t^k| \| A \|^{k} (k+1)2^{-t_1 m} \right. \]
\[ \cdot \left. (1+n2^{-t_1})^k + n \sum_{j=1}^{k} (1+n2^{-t_1})^{j-1} \right] \]

This is a complicated bound due to the dependence of round-off error from one \( \hat{b}_k \) to another. But for a certain class of matrices and with reasonable assumptions it can serve as a guide to the applicability of the method.
Let us now consider a special case of the general method, develop an eigenvalue recovery technique for it, and discuss its application and advantages. For this special case we take \( G(x) = e^{-x} \), although everything we say shall be analogously true for \( G(x) = e^x \), of course.

The idea behind this attack is to evaluate \( H(x) \) at equally spaced points \( t_0, t_1, \ldots, t_n \) where we define \( h \) such that \( h = t_i - t_{i-1} \) for \( i = 1, \ldots, n \). We then let \( H_j(t_i) \) denote the \( j \)th component of \( H(t_i) \) and form the following difference scheme:

\[
\begin{align*}
\sum_{i=0}^{n-1} c_i H_i(t_0) + \sum_{i=0}^{n-1} c_i H_i(t_1) + \ldots + c_{n-1} H_{n-1}(t_n) &= 0 \\
\sum_{i=0}^{n-1} c_i H_i(t_0) + \sum_{i=0}^{n-1} c_i H_i(t_1) + \ldots + c_{n-1} H_{n-1}(t_n) &= 0 \\
&\vdots \\
\sum_{i=0}^{n-1} c_i H_i(t_0) + \sum_{i=0}^{n-1} c_i H_i(t_1) + \ldots + c_{n-1} H_{n-1}(t_n) &= 0,
\end{align*}
\]

or in matrix form:

\[
H \hat{c} = \hat{0}.
\]

Remembering that

\[
H_j(t_i) = e^{-\lambda_1 t_i u_{1j}} + \ldots + e^{-\lambda_n t_i u_{nj}},
\]

we now show that this system always has a solution with \( c_n = 1 \), and that the \( c_i \)'s are the coefficients of the characteristic polynomial of \( e^{-hA} \).

**Theorem 1:**

If \( Y = \begin{bmatrix} 1 & y_1 & \cdots & y_1^{n-1} & y_1^n \\ 1 & y_2 & \cdots & y_2^{n-1} & y_2^n \\ \vdots \\ 1 & y_n & \cdots & y_n^{n-1} & y_n^n \end{bmatrix} \),

then \( Y e^{-hA} = Y \).

where

\[
Y e^{-hA} = \begin{bmatrix} 1 & y_1 & \cdots & y_1^{n-1} & y_1^n \\ 1 & y_2 & \cdots & y_2^{n-1} & y_2^n \\ \vdots \\ 1 & y_n & \cdots & y_n^{n-1} & y_n^n \end{bmatrix} = Y.
\]
then the rank of Y equals the number of distinct y_i. 

Proof: Let us make the following definitions:

\[ k(p)_p = (y_1 + \ldots + y_p), \quad k(p)_{p-1} = -\sum_{i_1=1}^{p-1} \left( \sum_{i_2 > i_1}^{p} (y_{i_1} y_{i_2}) \right), \]

\[ k(p)_{p-2} = \sum_{i_1=1}^{p-2} \left( \sum_{i_2 > i_1}^{p-1} \left( \sum_{i_3 > i_2}^{p} (y_{i_1} y_{i_2} y_{i_3}) \right) \right), \ldots, \]

\[ k(p)_1 = (-1)^{p-1} (y_1 \ldots y_p). \]

Let the above be defined for any positive integer p \in \mathbb{N}, and let \( k(p)_0 = 0 \).

Before proceeding further in this proof, we must establish two lemmas concerning the quantities \( k(m)_j \) and \( n \) defined above.

Lemma 1: \( k(m+1)_j = -y_{m+1} k(m)_j + k(m)_{j-1} \), for 
\[ 1 \leq j \leq m \leq n. \]

Proof: Intuitively \( |k(m)_j| \) is the sum of all products of the first \( m \) variables taken \((m-j+1)\) at a time, and is given by:

\[ k(m)_j = (-1)^{m-j} \sum_{i_1=1}^{j} \left( \sum_{i_2 > i_1}^{j+1} \left( \sum_{i_3 > i_2}^{m} (y_{i_1} \ldots y_{i_{m-j+1}}) \right) \right), \]

with similar formulas for \( k(m+1)_j \) and \( k(m)_{j-1} \).

Now \( |y_{m+1} k(m)_j| \) is the sum of all products of the first \((m+1)\) variables taken \((m-j+2)\) at a time except those which do not contain \( y_{m+1} \). But \( k(m)_{j-1} \) is the sum of all
products of the first \( m \) variables taken \((m-j+2)\) at a time. Thus 
\[ (-y_{m+1} k(m)_j + k(m)_{j-1}) \]
equals 
\[ (-1)^{m-j+1} \times \text{sum of all products of the first } (m+1) \text{ variables taken } (m-j+2) \text{ at a time, but this is exactly } k(m+1)_j. \]
Thus 
\[ -y_{m+1} k(m)_j + k(m)_{j-1} = k(m+1)_j. \]

Lemma 2:
\[ k(m)_1 y_1 + k(m+1)_2 y_2^2 + \ldots + k(m)_m y_m = y_{m+1} \]
for any \( i \) and \( m \) such that \( 1 \leq i \leq m \leq n \).

Proof by induction:

i.) \( q=1 \): \( k(1)_1 = y_1 \), then \( k(1)_1 y_1 = y_1^2 \).

\[ q=2: \] \( k(2)_1 y_1 + k(2)_2 y_2^2 = (-1)y_1 y_2 y_1 + (y_1 y_2) y_1^2. \)

If \( i=1 \): \[ -y_1^2 + y_2^3 + y_1^2 = y_1^3. \]

If \( i=2 \): \[ -y_1 y_2^2 + (y_1 y_2 + y_2^3) = y_2^3. \]

ii.) Consider any arbitrary \( q \) such that \( 1 \leq i \leq q < n \),
and \( k(q)_1 y_1 + k(q)_2 y_2^2 + \ldots + k(q)_q y_q \)
\[ k(q+1)_1 y_1 + k(q+1)_2 y_2^2 + \ldots + k(q+1)_q y_q + k(q+1)_{q+1} y_{q+1} \]
\[ = (-y_{q+1} k(q)_1) y_1 + (-y_{q+1} k(q)_2 + k(q)_1) y_1^2 + \ldots \]
\[ + (-y_{q+1} k(q)_q + k(q)_q y_q) y_1 + k(q+1)_{q+1} y_{q+1} \]
\[ = -y_{q+1} (k(q)_1 y_1 + k(q)_2 y_2^2 + \ldots + k(q)_q y_q) \]
\[ + k(q)_1 y_1^2 + \ldots + k(q)_q - k(q)_{q+1} (k(q)_q y_{q+1}) \]
\[ + k(q+1)q+1y_{q+1} \]
\[ = -y_{q+1}y_{q+1} + y_1(k(q)_1y_1 + \ldots + k(q)y_{q+1}) \]
\[ -(y_1 + \ldots + y_q)y_{q+1} + (y_1 + \ldots + y_q + y_{q+1})y_{q+1} \]
\[ = -y_{q+1}y_{q+1} + y_1y_{q+1} + y_{q+1}y_{q+1} \]
\[ = y_{q+2}. \]

Thus the proof of this lemma is complete.

We now proceed with the proof of the theorem. Let

\[
D = \begin{bmatrix}
y_1 & 0 & \ldots & 0 \\
0 & y_2 & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & \ldots & 0 & y_n
\end{bmatrix}
\]

Since no \( y_1 = 0 \), then \( D \) is non-singular. Let \( Y_1 = DY \), then \( Y_1 \) and \( Y \) have the same rank and

\[
Y_1 = \begin{bmatrix}
y_1 & y_2 & \ldots & y_n & y_{n+1} \\
y_1 & y_2 & \ldots & y_n & y_{n+1} \\
y_2 & y_2 & \ldots & y_n & y_{n+1} \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
y_n & y_2 & \ldots & y_n & y_{n+1}
\end{bmatrix}
\]

Let \( \hat{e}_1, \hat{e}_2, \ldots, \hat{e}_n, \hat{e}_{n+1} \) denote the columns of \( Y_1 \). Then, by the previous lemmas, \( k(n)\hat{e}_1 + k(n)\hat{e}_2 + \ldots + k(n)\hat{e}_n = \hat{e}_{n+1} \). Thus the rank of \( Y_1 \) equals the rank of \( Y_2 \), the matrix formed by deleting the \((n+1)^{st}\) column. But \( Y_2 \) equals \( D \) times a Vandermonde matrix formed from the \( y_1 \)'s, and this matrix
has a rank equal to the number of distinct $y_i$'s. Thus the rank of $Y$ equals the number of distinct $y_i$'s, and the theorem is proved.

Theorem 2: If $\hat{H}(x) = e^{-\lambda_1 \hat{u}_1} + \ldots + e^{-\lambda_n \hat{u}_n}$, where $\hat{u}_1, \ldots, \hat{u}_n$ are the $n$ independent eigenvectors of $A$ which are assumed to exist, then the above system, $H \hat{v} = \hat{0}$, has a solution with $c_n = 1$, and the rank of $H$ equals the number of distinct eigenvalues of $A$.

Proof: Let $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of $A$ and let $m \leq n$ of them be distinct. Let $x_1 = e^{-\lambda_1}$, and assume with no loss of generality that $x_1, \ldots, x_m$ are distinct. Also let $t_0 = p$.

Now the system has the form:

$$H \hat{v} = \begin{bmatrix} H_1(t_0) & H_1(t_1) & \ldots & H_{n-1}(t_{n-1}) \\ H_2(t_0) & H_2(t_1) & \ldots & H_{n-1}(t_{n-1}) \\ \vdots & \vdots & \ddots & \vdots \\ H_n(t_0) & H_n(t_1) & \ldots & H_{n-1}(t_{n-1}) \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-1} \end{bmatrix} = \begin{bmatrix} -H_1(t_n) \\ -H_2(t_n) \\ \vdots \\ -H_n(t_n) \end{bmatrix} = \hat{k}.$$
\[
\begin{bmatrix}
  u_{11} & u_{12} & \cdots & u_{1n} \\
  u_{21} & u_{22} & \cdots & u_{2n} \\
  \vdots & \vdots & \ddots & \vdots \\
  u_{n1} & u_{n2} & \cdots & u_{nn}
\end{bmatrix}
\begin{bmatrix}
  x_1^p & x_1^{p+h} & \cdots & x_1^{p+(n-1)h} \\
  x_2^p & x_2^{p+h} & \cdots & x_2^{p+(n-1)h} \\
  \vdots & \vdots & \ddots & \vdots \\
  x_n^p & x_n^{p+h} & \cdots & x_n^{p+(n-1)h}
\end{bmatrix}
= UX_p.
\]

Since \( U, \ldots, U_n \) span \( C^n \), \( U \) is non-singular and of rank \( n \).

Now
\[
X_p = \begin{bmatrix}
  x_1^p & 0 & \cdots & 0 \\
  0 & x_2^p & \cdots & 0 \\
  \vdots & \vdots & \ddots & \vdots \\
  0 & 0 & \cdots & x_n^p
\end{bmatrix}
= \begin{bmatrix}
  1 & x_1^h & \cdots & x_1^{(n-1)h} \\
  1 & x_2^h & \cdots & x_2^{(n-1)h} \\
  \vdots & \vdots & \ddots & \vdots \\
  1 & x_n^h & \cdots & x_n^{(n-1)h}
\end{bmatrix}
= D_p X.
\]

Since no \( x_i = 0 \), the rank of \( D_p \) equals \( n \). Also, under the transformation \( y_i = x_i^h \), we have
\[
X = \begin{bmatrix}
  1 & y_1 & \cdots & y_1^{n-1} \\
  1 & y_2 & \cdots & y_2^{n-1} \\
  \vdots & \vdots & \ddots & \vdots \\
  1 & y_n & \cdots & y_n^{n-1}
\end{bmatrix}.
\]

Thus, since \( X \) is a Vandermonde matrix, its rank equals \( m \), the number of distinct \( x_i \)'s. Thus the rank of \( H \) equals \( m \).

Now consider the augmented matrix, \( H_a \). By an analogous procedure as above and using theorem 1, we can show that the rank of \( H_a \) equals \( m \), the number of distinct \( x_i \)'s.

Therefore, since the rank of the augmented matrix equals the rank of the coefficient matrix, the system has a solution with \( c_n = 1 \) and the theorem is true.
We show the completion of our eigenvalue recovery scheme through the following corollary.

Corollary: If \( \{c_i\}_{i=0}^{n-1} \) and \( c_n = 1 \) is the solution of the system of equations in the previous theorem, then for any \( i \), \( y_i = e^{-\lambda_i h} = x_i^n \) is a root of the polynomial

\[
 c_0 + c_1 z + \ldots + c_{n-1} z^{n-1} + z^n = P(z).
\]

Proof: From \( H e = \hat{\kappa} \) of the previous proof we again factor to get

\[
 (UD_p X)e = -UD_p (x_1^n, x_2^n, \ldots, x_n^n)^T
\]

or

\[
 UD_p (Xe + (x_1^n, x_2^n, \ldots, x_n^n)^T) = \hat{\kappa}.
\]

(\( UD_p \)) is non-singular, so we multiply both sides by \( D_p^{-1} U^{-1} \) and get

\[
 \begin{bmatrix}
 1 & x^1 & \ldots & x_1^{(n-1)h} \\
 1 & x^2 & \ldots & x_2^{(n-1)h} \\
 \vdots & \vdots & \ddots & \vdots \\
 1 & x^n & \ldots & x_n^{(n-1)h}
\end{bmatrix}
 \begin{bmatrix}
 c_0 \\
 c_1 \\
 \vdots \\
 c_{n-1}
\end{bmatrix}
 +
 \begin{bmatrix}
 x_1^n \\
 x_2^n \\
 \vdots \\
 x_n^n
\end{bmatrix}
 = \hat{\kappa}.
\]

Looking at the \( i \)th component we see

\[
 c_0 + c_1 x_i^1 + \ldots + c_{n-1} x_i^{(n-1)h} + x_i^n = 0.
\]

Therefore \( x_i^1 = y_i = e^{-\lambda_i h} \) is a root of \( P(z) \).

We note here that \( P(z) \) is the characteristic equation of \( e^{-hA} \), and we shall hereafter call it the exponential characteristic equation of \( A \). As we noted in the introduction, iterative methods to find characteristic polynomials are not uncommon, but we shall show that this one is unique in that the exponential characteristic equation can be made
well-conditioned with respect to a perturbation in any of its coefficients.

Before we go into this, however, we pause to give an extension of this method which easily yields the eigenvectors once the eigenvalues are found if they are distinct.

Let
\[ \vec{y}_0 = \vec{R}(t_0), \vec{y}_1 = \vec{R}(t_0 + h), \ldots, \vec{y}_{n-1} = \vec{R}(t_0 + (n-1)h). \]

Choosing the first component of each of these vectors we get the following system of equations:

\[
\begin{align*}
\delta_{1}^{t_0} u_{11} + \cdots + \delta_{n}^{t_0} u_{n1} &= y_{01} \\
\delta_{1}^{t_0 + h} u_{11} + \cdots + \delta_{n}^{t_0 + h} u_{n1} &= y_{11} \\
\vdots & \quad \vdots \\
\delta_{1}^{t_0 + (n-1)h} u_{11} + \cdots + \delta_{n}^{t_0 + (n-1)h} u_{n1} &= y_{n-1,1}
\end{align*}
\]

or

\[
\begin{bmatrix}
\delta_{1}^{p} & \cdots & \delta_{n}^{p} \\
\delta_{1}^{p + h} & \cdots & \delta_{n}^{p + h} \\
\vdots & \ddots & \vdots \\
\delta_{1}^{p + (n-1)h} & \cdots & \delta_{n}^{p + (n-1)h}
\end{bmatrix}
\begin{bmatrix}
u_{11} \\
u_{12} \\
\vdots \\
u_{n1}
\end{bmatrix}
= \begin{bmatrix}y_{01} \\
y_{11} \\
\vdots \\
y_{n-1,1}
\end{bmatrix}
\]

Now, the coefficient matrix, \( X_p \), is the product of a diagonal matrix and a transformed Vandermonde matrix. Now, systems with Vandermonde coefficient matrices are common in physical problems, so we will merely outline the solution procedure here.

Using Cramer's rule we get
Now, the denominator determinant clearly equals

\[
\left( \prod_{k=1}^{n} x_k^p \right) \left( \prod_{k>j}^{n} (x_k^h-x_j^h) \right).
\]

Now expand the numerator determinant by the \(i\)th column. The first and last cofactor are of the above type, and the others can be evaluated by the following theorem.

This theorem is given without proof, but can be proved in exactly the same way as the Vandermonde determinant is commonly evaluated.

**Theorem 3:** If

\[
W = \begin{bmatrix}
1 & \cdots & 1 & 1 & \cdots & 1 \\
\vdots & & \vdots & & \vdots & \\
1_i & \cdots & w_{i-1} & w_{i+1} & \cdots & w_n \\
1_i+1 & \cdots & w_{i-1} & w_{i+1} & \cdots & w_n \\
\vdots & & \vdots & & \vdots & \\
1_i & \cdots & w_{i-1} & w_{i+1} & \cdots & w_n
\end{bmatrix}
\]
and \( z_1 = w_1, \ldots, z_{i-1} = w_{i-1} \) and \( z_i = w_{i+1}, \ldots, z_{n-1} = w_n \)

then \( \text{Det}(W) = \prod_{k>j}^{n-1} (z_k - z_j)^k(n-l)_r \),

where \( k(n-l)_r \) is, as in theorem 1, the sum of all of the products of the \((n-l)\) \( z_k \)'s taken \((n-r)\) at a time.

We will not go any further in this algebraic expansion except to note the large amount of cancellation of terms of the form \( \prod_{k>j}^{n} (x_k^h - x_j^h) \).

Thus the algebraic manipulations are greatly simplified. Lanczos (1956) develops an algorithm for solving systems of this type, and we refer the reader to him for further information.

We make a final note that the eigenvectors can be found by solving these systems for \( i=1, \ldots, n \).

We now proceed to the examination of the conditioning of the exponential characteristic equation.

Let us consider a polynomial \( f(z) = a_n z^n + \ldots + a_1 z + a_0 \) whose roots are \( z_1, \ldots, z_n \) and where \( z_r \) is a simple root. Assume that either due to round-off error or previous computation errors the polynomial is represented on the computer as \( F(z) = (a + \xi_1) z^n + \ldots + (a_1 + \xi_1) z + (a_0 + \xi_0) \), where each \( \xi_i \) need not be positive.
Let 
\[ g(z) = b_n z^n + \ldots + b_1 z + b_0 \]
be given where
\[ b_n = 1, \quad b_{n-1} = \frac{\xi_{n-1}}{\xi_n}, \ldots, \quad b_1 = \frac{\xi_1}{\xi_n}, \quad b_0 = \frac{\xi_0}{\xi_n} \]

Then with 
\[ \varepsilon = \xi_n \]
\[ F(z) = f(z) + \varepsilon g(z) \]

From Wilkinson (1963) and his reference to Goursat (1933), we have that one zero of \( F(z) \) can be given as:

\[ z_r(\varepsilon) = z_r + \sum_{k=1}^{\infty} p_k \varepsilon^k = z_r + h_1 \]

where 
\[ h_1 = \sum_{k=1}^{\infty} p_k \varepsilon^k \]

is certainly convergent for sufficiently small \( \varepsilon \), and, as we see below, the \( p_k \)'s are independent of \( \varepsilon \).

Substituting \( z_r(\varepsilon) \) into \( F(z) \), we get

\[ \sum_{j=0}^{n} c_j h_1^j + \varepsilon \sum_{j=0}^{n} d_j h_1^j \]

where
\[ c_j = f^{(j)}(z_r) \quad \text{and} \quad d_j = g^{(j)}(z_r) \]

Now \( c_0 \) is the only term in this expression that is not the coefficient of \( \varepsilon^k \), for some integer \( k \geq 1 \). Since \( c_0 = f(z_r) = 0 \), the expression can only be zero for all \( \varepsilon \) if the coefficient of \( \varepsilon^k \), for any integer \( k \), is zero.
Consider first the coefficient of $\varepsilon$. Here we have
\[ c_1p_1 + d = 0. \]

Thus
\[ p_1 = -\frac{d}{c_1} = -g(z_r) \frac{1}{f'(z_r)}. \]

Next consider the coefficient of $\varepsilon^2$. Here we have
\[ c_1p_2 + c_2p_1^2 + d_1p_1 = 0. \]

Then
\[ p_2 = -\frac{c_2p_1^2 + d_1p_1}{c_1} \]
\[ = 2g(z_r)g'(z_r)f'(z_r)-g^2(z_r)f''(z_r) \frac{1}{f'(z_r)^3}. \]

Continuing we see that we can choose each $p_k$ in this manner so that we have verified that $F(z_r(\varepsilon)) = 0$ and the $p_k$'s are independent of $\varepsilon$.

Let us assume that the root-finding technique we are using is "exact", i.e., accurate within round-off up to the last significant figure the computer can carry. Even in this somewhat idealistic case, the difference between the exact root $z_r$ and the computed root $z_r(\varepsilon)$ can be large. We see that
\[ |z_r - z_r(\varepsilon)| = \left| \sum_{k=1}^{\infty} p_k \varepsilon^k \right| \leq \sum_{k=1}^{\infty} |p_k| |\varepsilon|^k. \]

Then
\[ |z_r - z_r(\varepsilon)| \leq k_1 |\varepsilon|. \]
where \( K = |p_1| + |p_2| |\xi| + \sum_{k=3}^{\infty} |p_k| |\xi^{k-1}|. \)

Then \( K = |p_1| = \left| \frac{\xi(z_r)}{f'(z_r)} \right| \) as \( \xi \) becomes small.

The point here is that, even though \( \xi \) is very small, the constant \( K \) can be extremely large. This is especially true as another root of \( f(z) \) is taken very close to \( z_r \) making \( f'(z_r) \) close to zero and possibly making \( |p_1| \) very large.

There are, however, other cases of ill-conditioning for roots other than those that are extremely close together. Wilkinson (1963) carefully examines the twentieth degree polynomial with roots

\[ z_1 = 20, z_2 = 19, \ldots, z_{20} = 1 \]

by examining a perturbation of magnitude \( \xi \) separately in each of the coefficients \( a_k; k=0,1,\ldots,19 \). Here for \( z_5 = 16 \) and \( k=15 \) he finds a maximum perturbation in the zeros.

\[ \delta z_r \equiv |z_r - z_r(\xi)| = \xi(3.7 \times 10^4), \]

which shows that this polynomial is very ill-conditioned indeed. In this example the absolute difference in the roots is one, but the ratio is \( (k+1)/k \).

In another example where \( z_r = 2^{-r} \) for \( r = 1, \ldots, 20 \) the ratio is two, but the absolute difference is very small when \( r \) approaches 20. However, it turns out that this polynomial is well-conditioned and the smallest roots are the most stable, indicating that the ratio of the roots is more important than
their absolute difference in affecting the conditioning. This is further substantiated in the work that follows.

Let us consider these two examples of Wilkinson's within the framework of our eigenvalue method with \( G(x) = e^{-x} \), and assume that the matrix \( A \) has eigenvalues \( \lambda_1 = 20, \ldots, \lambda_{20} = 1 \). Here our method yields the polynomial with roots \( e^{-rh} \) instead of \( r \) for \( r = 1, 2, \ldots, 20 \). The absolute difference of these roots is small as \( r \) approaches 20, but their ratio is \( e^h \). Thus making \( h \) larger increases this ratio. In particular, if we choose \( h = \ln 2 \), then our roots become \( 2^{-r} \) and this polynomial is the second example examined by Wilkinson and is well-conditioned with respect to a perturbation in any one of its coefficients.

In general, for an arbitrary positive \( h \) and a polynomial with real roots, we use the relation that, if \( e^{-zr^h} \) is simple,

\[
\sigma'(e^{-zr^h}) = -\xi \frac{g(e^{-zr^h})}{f'(e^{-zr^h})}.
\]

We consider \( \sigma'_{ak} = \xi \) separately for each integer \( k \) between one and twenty. Thus the \( g(z) \) introduced at the beginning of this conditioning analysis equals \( z^k \), and

\[
|\sigma'(e^{-zr^h})| = \left| \frac{\xi e^{-zr^hk}}{\prod_{j=1}^{20} (e^{-zr^h} - e^{-zh})} \right| = |a(h)|.
\]

For best conditioning we want \( |\sigma'(e^{-zr^h})| \), and thus \( |a(h)| \).
to be as small as we desire with a proper choice of \( h \) to make the perturbation of the root \( e^{-z r h} \) as small as possible. Hereafter let us take this as what we mean by "well-conditioned". Let \( a_{20} = 1 \) so there is no perturbation in this coefficient.

First let us assume that there is an even number of \( z_j \)'s which are less than \( z_r \). Then, since \( h > 0 \), \( a(h) > 0 \).

Let

\[
k_{rj} = \begin{cases} 
1 & \text{if } z_r < z_j \\
-1 & \text{if } z_r > z_j
\end{cases}
\]

Then

\[
\ln(a(h)) = -z_r h k - \ln \left( \prod_{r \neq j=1}^{20} (e^{-z_r h} - e^{-z_j h}) \right)
\]

\[
= -z_r h k - \sum_{r \neq j=1}^{20} \ln \left[ k_{rj} (e^{-z_r h} - e^{-z_j h}) \right].
\]

Then

\[
\frac{d(\ln(a(h)))}{dh} = -k z_r - \sum_{r \neq j=1}^{20} \left[ \frac{-z_r e^{-z_r h} + z_j e^{-z_j h}}{e^{-z_r h} - e^{-z_j h}} \right]
\]

\[
= -k z_r + \sum_{r \neq j=1}^{20} \left( z_r + \frac{(z_r - z_j) e^{-z_j h}}{e^{-z_r h} - e^{-z_j h}} \right)
\]

\[
= (19-k) z_r + \sum_{r \neq j=1}^{20} \frac{(z_r - z_j) e^{-z_j h}}{e^{-z_r h} - e^{-z_j h}}.
\]

Now if \( z_r - z_j > 0 \) then \( e^{-z_r h} - e^{-z_j h} < 0 \), and vice versa. Thus all of the terms in the summation are negative.
If \( k=19 \) or if \( k<19 \) and \( z_\tau<0 \), then the entire expression is negative, and \( \ln(a(h)) \) has a negative slope for all \( h>0 \) which never approaches zero, and thus decreases for increasing \( h \) such that \( \lim_{h \to \infty} \ln(a(h)) = -\infty \). Thus \( a(h) \) is also a decreasing function of \( h \), and we can make the perturbation in the root \( e^{-z_\tau h} \) as small as we desire by making \( h \) large, since \( \lim_{h \to \infty} a(h) = 0 \). This means that \( e^{-z_\tau h} \) can be made well-conditioned in this case. There are, of course, practical limitations on the size of \( h \) due to round-off and truncation error in forming \( \hat{h}(x) \).

If we have the other case where \( k<19 \) and \( z_\tau>0 \), the leading term of \( \frac{d}{dh} (\ln(a(h))) \) is positive and it seems possible that \( \frac{d}{dh} (\ln(a(h))) \) is zero. The leading term in the summation is the only positive term, and it does not depend on \( h \). Let us consider then, the negative summation and consider its magnitude as a function of \( h \).

Now
\[
\lim_{h \to \infty} \sum_{r \neq j=1}^{20} \frac{(z_r - z_j) e^{-z_j h}}{e^{-z_r h} - e^{-z_j h}} = \lim_{h \to \infty} \sum_{r \neq j=1}^{20} \frac{(z_r - z_j) e^{(z_r - z_j) h}}{1 - e^{(z_r - z_j) h}}
\]
\[
= \sum_{r \neq j=1}^{20} \lim_{h \to \infty} \frac{(z_r - z_j) e^{(z_r - z_j) h}}{1 - e^{(z_r - z_j) h}},
\]
where the last equality holds if each of the individual limits exists. Each of these has the form:
\[
\lim_{h \to \infty} \frac{\alpha e^{\alpha h}}{1-e^{\alpha h}} = \lim_{h \to \infty} \left( -\alpha + \frac{\alpha}{1-e^{\alpha h}} \right) = \begin{cases} 0 & \text{if } \alpha < 0 \\ -\alpha & \text{if } \alpha > 0. \end{cases}
\]

Thus
\[
\lim_{h \to \infty} \sum_{r \neq j=1}^{20} \frac{(z_r - z_j)e^{-z_j h}}{e^{-z_r h} - e^{-z_j h}} = \sum_{r \neq j=1}^{20} w_j
\]

where

\[
w_j = \begin{cases} 0 & \text{if } z_r < z_j \\ (z_j - z_r) & \text{if } z_r > z_j. \end{cases}
\]

Now if \( \left| \sum_{r \neq j=1}^{20} w_j \right| > (19-k)z_r \), then \( \frac{d}{dh} (\ln(a(h))) \)
ultimately has a negative slope and the same arguments as above show well-conditioning.

Let us consider the case where \( \left| \sum_{r \neq j=1}^{20} w_j \right| < (19-k)z_r \)
in the following manner.

Let

\[
b_j(h) = \frac{(z_r - z_j)e^{-z_j h}}{e^{-z_r h} - e^{-z_j h}} < 0,
\]

and for any \( h > 0 \), let us consider its critical points.

\[
b'_j(h) = \frac{-(z_r - z_j)e^{-z_j h}(e^{-z_r h} - e^{-z_j h}) - (z_r - z_j)e^{-z_j h}(-z_r e^{-z_r h} + z_j e^{-z_j h})}{(e^{-z_r h} - e^{-z_j h})^2}
\]

\[
= \frac{-(z_r - z_j)e^{-z_j h}}{(e^{-z_r h} - e^{-z_j h})^2} \left( z_j e^{-z_r h} - z_r e^{-z_j h} \right) + (z_r - z_j)^2 e^{-z_j h} e^{-z_r h}
\]

\[
= \frac{(z_r - z_j)^2 e^{-z_r h} - e^{-z_j h}}{(e^{-z_r h} - e^{-z_j h})^2} > 0.
\]
Thus for positive $h$, $b(h) = \sum_{r \neq j=1}^{20} b_j(h)$ has no critical points, and since $b'(h) > 0$, $b(h)$, and thus $\frac{d}{dh} (\ln(a(h)))$, is an increasing function of $h$. Now we have in this particular case,

$$\lim_{h \to \infty} \frac{d}{dh} (\ln(a(h))) = (19-k)z_r + \sum_{r \neq j=1}^{20} w_j > 0.$$ 

Also, whenever we have an even number of $z_j$'s less than $z_r$, we have $\lim_{h \to 0} \ln(a(h)) = \infty$ and $\lim_{h \to 0} \frac{d}{dh} (\ln(a(h))) = -\infty$.

Since $\frac{d}{dh} (\ln(a(h))) = (19-k)z_r + b(h)$ and $b(h)$ is an increasing function of $h$, then there exists one and only one value $h_0$ where $\frac{d}{dh} (\ln(a(h)))$ equals zero. Thus $\ln(a(h))$ and $a(h)$ have their only minimum at $h = h_0$. Thus the polynomial for $(19-k)z_r + \sum_{r \neq j=1}^{20} w_j > 0$ is "best-conditioned",

\[ \text{i.e., } a(h) \text{ is minimized, if not well-conditioned at } h = h_0. \]

Let us pause to note here that one case where

\[ (19-k)z_r = \left| \sum_{r \neq j=1}^{20} w_j \right| \]

has not been discussed. Here the slope of $\ln(a(h))$ becomes zero as $h$ approaches infinity and $\ln(a(h))$ has a horizontal asymptote. Since $b(h)$ is increasing, the polynomial becomes "best-conditioned" in the limit
as h becomes large, but not necessarily well-conditioned.

Now let us consider the remaining case where there is an odd number of \( z_j \) less than \( z_r \). Let \( z_s \) be one such value of \( z_j \).

Then

\[
\left| a(h) \right| = \frac{1}{e^{-z_sh} - e^{-z_{rh}}} \cdot \frac{e^{-z_r kh}}{\prod_{j=1 \atop j \neq r, s} (e^{-z_{rh}} - e^{-z_{jh}})}
\]

and again

\[
\begin{align*}
\text{let } k_{rj} & = \begin{cases} 
1 & \text{if } z_r < z_j \\
-1 & \text{if } z_r > z_j
\end{cases} \\
\text{Then } \ln \left| a(h) \right| & = -z_r kh - \ln(e^{-z_{sh}} - e^{-z_{rh}}) \\
& - \sum_{j=1 \atop j \neq r, s}^{20} \ln(k_{rj}(e^{-z_{rh}} - e^{-z_{jh}})),
\end{align*}
\]

and

\[
\frac{d}{dh} \ln \left| a(h) \right| = -z_r k + \frac{z_s e^{-z_{sh}} - z_r e^{-z_{rh}}}{(e^{-z_{sh}} - e^{-z_{rh}})}
\]

\[
+ \sum_{j=1 \atop j \neq r, s}^{20} \frac{k_{rj}(z_r e^{-z_{rh}} - z_r e^{-z_{jh}})}{k_{rj}(e^{-z_{rh}} - e^{-z_{jh}})}
\]

\[
= z_r k + z_r + \frac{(z_s - z_r) e^{-z_{sh}}}{e^{-z_{sh}} - e^{-z_{rh}}}
\]

\[
+ \sum_{j=1 \atop j \neq r, s}^{20} \left( z_r + \frac{(z_r - z_j) e^{-z_{jh}}}{e^{-z_{rh}} - e^{-z_{jh}}} \right)
\]
\[(19-k)z^r + (z_r - z_s) e^{-z^h} + \sum_{j=1, j\neq r,s}^{20} (z_r - z_j) e^{-z^h} \]

Since this derivative is now the same as \(\frac{d(\ln(a(h)))}{dh}\) in the previous case, the results of the analysis of \(|a(h)|\) in this case are the same as the results in the previous analysis of \(a(h)\).

Now we pause to summarize this perturbation analysis, regardless of the number of \(z_j\)'s less than \(z_r\).

1.) If \(z_r \leq 0\) or if \(z_r > 0\) and \((19-k)z_r + \sum_{r \neq j=1}^{20} w_j < 0\), then \(a(h)\) is a decreasing function of \(h\) and \(\lim_{h \to \infty} |a(h)| = 0\).

2.) If \((19-k)z_r + \sum_{r \neq j=1}^{20} w_j = 0\), then \(|a(h)|\) has a horizontal asymptote, and \(\lim_{h \to \infty} |a(h)| = K\), some positive constant.

3.) If \((19-k)z_r + \sum_{r \neq j=1}^{20} w_j > 0\), then \(|a(h)|\) is an increasing function of \(h\) and \(\lim_{h \to \infty} |a(h)| = \infty\). But there exists a single point, \(h = h_0\), where \(|a(h)|\) is at its minimum.
In all three cases \( \lim_{h \to 0} |a(h)| = \infty \). These cases are shown in Figure 1, for an even number of \( z_j < z_r \). When there is an odd number of \( z_j \) less than \( z_r \), \( a(h) < 0 \), and the graphs represent \( |a(h)| \).

Thus we see that for \( G(x) = e^{-x} \) and making \( h \) large, the exponential characteristic equation of \( A \) can always be made well-conditioned if the eigenvalues of \( A \) are real and non-positive, and can sometimes be well-conditioned when part or all of the eigenvalues are real and positive. Thus we note here that if \( \nu \) is a bound on the spectral radius of \( A \) and \( D = \mu I \), then the matrix \( A - D \) will have a well-conditioned exponential characteristic equation for sufficiently large \( h \).

It is clear that for \( G(x) = e^{x} \), the very same analysis of conditioning can be used with the roles of positive and negative \( z_r \) reversed. Thus the exponential characteristic polynomial of a matrix with non-negative eigenvalues can always be made well-conditioned for sufficiently large \( h \), and sometimes be made well-conditioned otherwise.

The above analysis was done with a twentieth degree polynomial to correspond to the work of Wilkinson (1963), but there is nothing in the analysis to restrict the degree to twenty.

Lastly, Wilkinson (1963) contends that the size of the
Figure 1. Magnitude of $a(h)$. Upper graph is of $a(h)$ for $z_r \leq 0$ or $z_r > 0$ and $(19-k)z_r + \sum_{r \neq j=1}^{20} w_j < 0$. Middle graph is of $a(h)$ for $(19-k)z_r + \sum_{r \neq j=1}^{20} w_j = 0$. Bottom graph is of $a(h)$ for $(19-k)z_r + \sum_{r \neq j=1}^{20} w_j > 0$. 
The ratio of the roots is more important than their absolute difference with respect to the condition of a polynomial, i.e., the closer the ratio of the larger root in magnitude over the smaller root is to one, the worse the conditioning. The above analysis supports this contention.

For example, consider $z_j < z_r < 0$. We have shown conditioning is improved by increasing $h$ and choosing $G(x) = e^{-x}$. Now even though $\frac{z_j}{z_r} = 1$, $\frac{e^{-z_j h}}{e^{-z_r h}} = e^{(z_r - z_j) h}$ and increasing $h$ obviously increases this ratio.

In the case where $0 < z_j < z_r$ we have shown for $G(x) = e^{x}$, conditioning improves as $h$ increases. Once again, even though $\frac{z_r}{z_j} = 1$, $\lim_{h \to \infty} e^{z_r h} = \lim_{h \to \infty} e^{(z_r - z_j) h} = \infty$.

Thus the above contention is supported.

One further note about this method is that a matrix with real roots can always be translated, as shown above, to make its eigenvalues have the same sign by adding or subtracting a diagonal matrix of the form $\mu \cdot I$, where $\mu > \rho(A)$. We have also recommended a division of this resulting matrix by a bound $\gamma$ on its spectral radius. However, it is often true that a close bound $\gamma$ is difficult to obtain. One way to approach this is to choose what seems to be a reasonable $\gamma$ and start building the vector sequence $\{b_k\}$. It is usually
easy to see within a few iterations if this sequence is converging to \( \sigma \). If not, choose \( \gamma \) a bit larger and repeat. This requires a certain amount of computation, but often gives a more practical bound than a theoretical approach. For instance, in Example 4 below, the Gershgorin bound on \( \rho(A) \) is 931, but \( \rho(A) \) itself equals 13.

We have not considered multiple roots here, but Wilkinson (1963) shows that there is the same "power-series-in-\( \varepsilon \)" form of root for the machine representation of the polynomial, and that the same type of analysis will hold. Complex roots are not discussed here either, but if their real parts are not zero, then either the ratio \( \frac{e^{-z_j h}}{e^{-z_r h}} \), if \( \text{Re}(z_j) < \text{Re}(z_r) < 0 \), or the ratio \( \frac{e^{z_r h}}{e^{z_j h}} \), if \( 0 < \text{Re}(z_j) < \text{Re}(z_r) \), increases as \( h \) increases. The conditioning in this case is again exhibited by Example 4, which shows that as \( h \) goes from 1/3 to 3 that the accuracy of the eigenvalues steadily improves.

We lastly pause to note that there can be other ways of evaluating \( \tilde{H}(x) \), other than by a Maclaurin series expansion. In this case with \( G(x) = e^x \), for example, consider the differential system:
\[
\frac{dy_1}{dt} = a_{11}y_1 + \ldots + a_{1n}y_n \\
\frac{dy_2}{dt} = a_{21}y_1 + \ldots + a_{2n}y_n \\
\quad \vdots \\
\frac{dy_n}{dt} = a_{n1}y_1 + \ldots + a_{nn}y_n,
\]
or
\[
y = Ay,
\]
where \( A = (a_{ij}) \).

The solution of this system of differential equations is
\[
\hat{y}(t) = e^{tA}\hat{y}_0,
\]
where \( \hat{y}(0) = \hat{y}_0 \).

Now
\[
\hat{H}(t) = e^{\lambda_1 t}\hat{u}_1 + \ldots + e^{\lambda_n t}\hat{u}_n = \sum_{k=0}^{\infty} \frac{t^k}{k!} \hat{b}_k
\]
\[
= \left( \sum_{k=0}^{\infty} \frac{t^k}{k!} A^k \right) \hat{b}_0
\]
\[
= e^{tA}\hat{b}_0,
\]
where
\[
e^B = \lim_{n \to \infty} \left( I + B + \frac{B^2}{2} + \ldots + \frac{B^n}{n!} \right),
\]
and exists for all \( B \).

Thus we can use an iterative technique such as Runge-Kutta to solve the system \( \hat{y} = A\hat{y} \) and \( \hat{y}(0) = \hat{b}_0 \) at the points \( t_0, t_0+h, \ldots, t_0+(n-1)h; \) and avoid having to use a Maclaurin vector expansion.

For other techniques involving other choices for \( G(x) \), this approach may be advantageous, depending on the character of the system of differential equations that must be used.
We now exhibit some examples that were done using $G(x) = e^{ix}$.

Example 1:

$$A = \begin{bmatrix} -0.5 & 2.5 & 0.5 & 0 \\ 0.5 & 1 & 0.5 & 0.5 \\ 0.5 & -1 & -0.5 & 0.5 \\ 0.5 & -1.5 & 0.5 & 0 \end{bmatrix}$$

The eigenvalues of $A$ are $\pm 1$, and $\pm 1/2$. The computed values were:

$$\lambda_1 = 0.999998 \quad \lambda_2 = -0.999998 \quad \lambda_3 = 0.500001 \quad \lambda_4 = -0.499999.$$  

This example illustrates a problem with eigenvalues of equal moduli.

Example 2:

$$A = \begin{bmatrix} 1 & 0 & 0.5 & 0.5 \\ 0 & 0.5 & 0 & 0 \\ -0.5 & 0 & 0 & -0.5 \\ 0.5 & 0 & 0.5 & 0 \end{bmatrix}$$

The eigenvalues of $A$ are $\lambda_1 = 1$ and $\lambda_2 = \lambda_3 = \lambda_4 = 1/2$. The computed values were

$$\lambda_1 = 1.000000 \quad \lambda_2 = \lambda_3 = \lambda_4 = 0.499999.$$ 

This example illustrates a triple eigenvalue which is not dominant.

Example 3:

$$A = \begin{bmatrix} 0 & 0.0401 & 0 & -0.000004 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$
The eigenvalues of $A$ are $\pm 0.1$ and $\pm 0.2$. The computed values were:

$$\lambda_1 = 0.200001 \quad \lambda_2 = -0.199998 \quad \lambda_3 = 0.010006 \quad \lambda_4 = -0.010008.$$  

This example illustrates a problem with "close" eigenvalues and a matrix whose determinant equals 0.000004 and is thus "close" to being singular.

Example 4:

Our last example is treated more extensively. Here

$$A = \begin{bmatrix}
-31 & 96 & 84 & -96 & -240 & 384 \\
-30 & 74 & 45 & -60 & -150 & 240 \\
-12 & 6 & -25 & 24 & 60 & -95 \\
-12 & 18 & -6 & 43 & 85 & -145 \\
-36 & 54 & -18 & 66 & 102 & -174 \\
-24 & 36 & -12 & 50 & 80 & -137
\end{bmatrix}.$$  

The eigenvalues of $A$ are $5 \pm 12i$, $8 \pm 6i$, and $9i$. We illustrate the increased accuracy of the computed values due to increasing $h$ in the following table. One important consideration here is that each evaluation took approximately 35 seconds, regardless of $h$, or the accuracy of the computation. Also the polynomial root-finder on the machine was in single precision, so six places is nearing the limitation of its accuracy.
<table>
<thead>
<tr>
<th>$h$</th>
<th>$\lambda_1$ and $\lambda_2$</th>
<th>$\lambda_3$ and $\lambda_4$</th>
<th>$\lambda_5$ and $\lambda_6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/3</td>
<td>$4.771800 \pm 12.136631 , i$</td>
<td>$8.548610 \pm 6.265285 , i$</td>
<td>$0.085026 \pm 8.448740 , i$</td>
</tr>
<tr>
<td>2/3</td>
<td>$4.999195 \pm 11.985877 , i$</td>
<td>$8.011786 \pm 6.016838 , i$</td>
<td>$0.001115 \pm 8.995581 , i$</td>
</tr>
<tr>
<td>1</td>
<td>$5.000927 \pm 11.999351 , i$</td>
<td>$7.997694 \pm 6.001117 , i$</td>
<td>$0.000078 \pm 8.999323 , i$</td>
</tr>
<tr>
<td>4/3</td>
<td>$5.000173 \pm 12.000105 , i$</td>
<td>$8.000622 \pm 6.000146 , i$</td>
<td>$0.000021 \pm 9.000089 , i$</td>
</tr>
<tr>
<td>5/3</td>
<td>$4.999987 \pm 12.000033 , i$</td>
<td>$8.000026 \pm 5.999941 , i$</td>
<td>$0.000034 \pm 9.000074 , i$</td>
</tr>
<tr>
<td>2</td>
<td>$4.999977 \pm 12.000007 , i$</td>
<td>$9.000018 \pm 5.999967 , i$</td>
<td>$0.000019 \pm 9.000017 , i$</td>
</tr>
<tr>
<td>3</td>
<td>$5.000000 \pm 12.000000 , i$</td>
<td>$7.999998 \pm 6.000000 , i$</td>
<td>$0.000004 \pm 8.999994 , i$</td>
</tr>
</tbody>
</table>
OTHER POWER SERIES TECHNIQUES

The second recovery technique we consider illustrates the principle of choosing \( G(t) \) to emphasize the eigenvalues one at a time. Here we let \( G(x) = \frac{1 - \cos m(x-1)}{(x-1)^2} \), for some arbitrary positive \( m \). For simplicity let us analyze \( G_1(t) = \frac{1-\cos mt}{t^2} \), where \( t = x-1 \).

Now \( \lim_{t \to 0} G_1(t) = \frac{m^2}{2} \), \( G_1'(t) = \frac{mt \sin mt - 2(1-\cos mt)}{t^3} \), and \( G_1''(t) = \frac{(m^2 t^2 - 6) \cos mt - 4mt \sin mt + 6}{t^4} \).

Now \( G_1'(0) = 0 \) and \( G_1''(0) = -m^4 < 0 \), thus \( t = 0 \) is a maximum of \( G_1(t) \).

We now search for the other maxima and minima and examine their magnitude and distance from the one at \( t = 0 \). Then in what follows \( t \neq 0 \). If \( G_1'(t) = 0 \), we have

\[
mt \sin mt - 2(1-\cos mt) = 0,
\]
or

\[
u \sin u - 2 + 2 \cos u = 0.
\]
Obviously \( u = 2n\pi \) is a solution and \( u = (2n-1)\pi \) is not, for \( n=1,2,3,... \).

Let us consider \( u \in (0,\pi) \), then

\[
u = 2 \frac{1-\cos u}{\sin u} = 2 \tan \frac{u}{2},
\]
or

\[v = \tan v, \text{ where } v = u/2.\] Now \( v \in (0, \frac{\pi}{2}) \) and in this interval the equation is satisfied by no value of \( v \), since \( v=0 \) is a solution and \( \sec^2 v > 1 \).
The first positive solution of this equation comes in the interval \((\pi, 3\pi/2)\), since \(\tan v < 0\) for \(v \in (\pi/2, \pi)\). This solution is found to be approximately \(v_1 = 4.493\). In terms of \(m\) and \(t\), this is \(t_1 = 2v_1/m\) and \(t_1 \in (2\pi/m, 3\pi/m)\). Now \(t' = 2\pi/m\) is the first critical point to the right of \(t=0\) and it represents a minimum for \(G_1(t)\). Since \(t_1\) is the second critical point to the right of \(t=0\), there are no maxima in the interval \((0, t_1)\). Now, since \(v = \tan v\) has only one solution in every interval of length \(\pi\), say, \(v \in (n\pi, (n+1)\pi)\) and \(G_1(t)\) has minima at \(t=2n\pi/m\), then \(G_1(t)\) has only one other critical point in each interval \(t \in (2n\pi/m, 2(n+1)\pi/m)\), and this must be a maximum.

Theorem 4: If \(0 = t_0 < t_1 < t_2 < \ldots\) are maximum points of \(G_1(t)\), then \(G_1(0) = G_1(t_0) < G_1(t_1) < G_1(t_2) < \ldots\), for \(m > (4v_1)^{1/3} > 0\).

Proof: Let \(I_n = (2n\pi/m, 2(n+1)\pi/m)\), and we see from above that there is only one maximum in each \(I_n\). Now \(G_1(t) \geq 0\) and equals zero when \(t = 2n\pi/m; n=1,2,3,\ldots\). Thus \(|G_1(t)|\) is largest in any \(I_n\) when \(t\) is a maximal point of \(G_1(t)\). Let \(t_n\) and \(t_{n+1}\) be two adjacent maximal points of \(G_1(t)\) lying in \(I_n\) and \(I_{n+1}\), respectively. Then

\[
G_1(t_n) \geq G_1\left(\frac{(2n+1)\pi}{m}\right) = \frac{2m^2}{(2n+1)^2\pi^2} > \frac{2m^2}{(2n+2)^2\pi^2}
\]

and

\[
\geq \sup_{t \in I_{n+1}} \left(\frac{(1-\cos mt)}{(2n+2)^2\pi^2}\right)
\]
\[ \sup_{t \in \mathbb{I}_{n+1}} \left( \frac{1 - \cos mt}{t^2} \right) = G_1(t_{n+1}). \]

Now \( G_1(0) = m^2/2 \) and \( G_1(t_1) = 2v_1/m = \frac{2 \cdot 4.493}{m} \).

Then \( G_1(0) - G_1(t_1) = m^2/2 - 2v_1/m > 0 \) for \( m > (4v_1)^{1/3} > 0 \).

Thus the theorem holds.

Now \( G_1(0) \) is of the order of \( m^3 G_1(t_1) \) as \( m \) becomes large. But making \( m \) large complicates this technique in two ways. First of all, the larger \( m \) is, the more terms that must be taken in the Maclaurin series, and since the Maclaurin coefficients are difficult to obtain, this is a major factor. Secondly, making \( m \) larger brings the maximal points of \( G(t) \) closer together. However, this last property also makes the slope of \( G_1(t) \) steeper for \( t \in (0, 2\pi/m) \), which we shall see is to our advantage. Lastly, since \( G_1(t) \) is even, we need not analyze it for negative values of \( t \).

Now, with \( t = x-1 \) and \( G_1(x-1) = G(x) \), we have

\[ H(x) = \frac{1 - \cos m(\lambda_1 x - 1)}{\lambda_1^2} \hat{u}_1 + \ldots + \frac{1 - \cos m(\lambda_n x - 1)}{\lambda_n^2} \hat{u}_n. \]

Each component, \( H_i(x) \); \( i=1, \ldots, n \); has dominant maxima at \( x=1/\lambda_j \); \( j=1, \ldots, n \); if we have a "sufficient spread" of the eigenvalues.

Let us discuss what we mean by a "sufficient spread". We refer to theorem 4 and consider \( G(\lambda_1 x) \) with \( m \) large enough that the dominant maximum occurs at \( x=1/\lambda_1 \). The next maximum to the right is \( x = t_{1+1} / \lambda_1 \). The first minimum to the right of
x = \frac{1}{\lambda_1} \text{ occurs at } x = \frac{2\pi + m}{\lambda_1}. \\

Thus for each \( \lambda_1 \), \( G(\lambda_1 x) \) goes from its dominant maximum to zero in the interval \( \left[ \frac{1}{\lambda_1}, \frac{2\pi}{\lambda_1} + \frac{1}{\lambda_1} \right] \), whose length is \( 2\pi/\lambda_1 \). Thus if two eigenvalues, \( \lambda_1 \) and \( \lambda_j \) are a distance of \( 2\pi/\lambda_1 \) apart or more, their dominant maxima will not interfere with each other. Now, for any matrix \( A \), the \( \lambda_1 \) and \( \lambda_j \) are fixed, but the \( m \) can be made larger to lessen the interference distance between the two principal maxima of \( G(\lambda_1 x) \) and \( G(\lambda_j x) \). We let \( S_m(\lambda_1, \lambda_j) = \left[ \frac{2\pi}{m} (\lambda_1 + \lambda_j) \right] \), and define it to be the "significance spread" of the two eigenvalues \( \lambda_1 \) and \( \lambda_j \). We note that \( \lambda_1 \) and \( \lambda_j \) can certainly be closer together than \( S_m(\lambda_1, \lambda_j) \) and still have distinguishable maxima at \( \frac{1}{\lambda_1} \) and \( \frac{1}{\lambda_j} \).

Let \( c_i \) and \( c_j \) be the coefficients of \( G(\lambda_i x) \) and \( G(\lambda_j x) \) in \( H_k(x) \), for some \( k \leq n \). Then \( S_m(\lambda_i, \lambda_j) \) is not affected, but the magnitudes of the principal maxima are \( c_i m^2/2 \) and \( c_j m^2/2 \), respectively. Thus the values of \( c_i \) and \( c_j \) can diminish the dominant peaks, but they must also diminish the secondary peaks, and the relative sizes of peaks for each eigenvalue are the same. However, \( c_i \) can make the dominant peak for \( \lambda_i \) smaller than some secondary peaks of \( G(\lambda_j x) \), and \( \lambda_i \) will not be distinguishable. Making \( m \) larger or choosing a different starting vector can help to overcome this difficulty.

To avoid a great many evaluations of \( \hat{H}(x) \), two approaches
may be used. First, obtaining an upper bound for $\rho(A)$ and a lower bound for the smallest eigenvalue in absolute value will limit the size of the interval to be examined.

Secondly, evaluating first at only a few points to determine the approximate eigenvalue location before doing an extensive search at these locations helps to limit the work.

One further note is, that if we form the series

$$\sum_{k=0}^{N} a_k x^{-k} b_k,$$

we get

$$\hat{H}(x) = G(\lambda_1/x) \hat{u}_1 + \ldots + G(\lambda_n/x) \hat{u}_n.$$

Each component here will have relative maxima at $x=\lambda_i$ instead of $x=1/\lambda_i$.

An example illustrating this technique, with the former series for $\hat{H}(x)$, is given by Figure 2. Here the eigenvalues of $A$ are $\pm 1$ and $\pm 1/2$. 
Figure 2. The power series technique with \( G(x) = \frac{1 - \cos m(x-1)}{(x-1)^2} \) where \( m = \) and the eigenvalues are \( \pm 1 \) and \( \pm 1/2 \).
Lastly we establish a technique which is based on a principle that Faddeev and Faddeeva (1963) call "component suppression." We assume here that the eigenvalues of $A$ are real.

In using this technique we also assume that an eigenvalue $\lambda_i$ of $A$ is known to be in some interval $(b,c)$, and all other eigenvalues of $A$ lie in the union of intervals $(a,b) \cup (c,d)$, where $a < b < c < d$. We consider the function

$$G(t) = \begin{cases} 1 & \text{for } t \in (b,c) \\ 0 & \text{for } t \in (c,d) \end{cases}$$

Let us now form some power series approximation to $G(t)$. For example, let us consider a Legendre series approximation where $a=-1$ and $d=1$. Then

$$G(t) = \sum_{n=0}^{\infty} A_n P_n(t), \quad \text{where } A_n = \frac{2n+1}{2} \int_{-1}^{1} G(x) P_n(x) \, dx$$

and $P_n(x)$ is the Legendre polynomial of degree $n$. With the particular $G(t)$ chosen above, we have

$$A_n = \frac{2n+1}{2} \sum_{j=0}^{\lfloor n/2 \rfloor} (-1)^j \frac{(2n-j)!}{j! (n-2j+1)! (n-j)!} (c^{n-2j+1} - b^{n-2j+1})$$

Now each $P_n(x) = \frac{(2n-1)x P_{n-1}(x) - (n-1) P_{n-2}(x)}{n}$, $n > 1$.

This formula is from Churchill (1963), as is the next:

$$P'_{n+1}(x) - P'_{n-1}(x) = (2n+1) P_n(x).$$
We can also calculate the $A_k$'s from this formula as follows:

$$A_k = \frac{2^{k+1}}{2} \int_b^c P_k(x)dx = \frac{1}{2} \int_b^c (P'_{k+1}(x) - P'_k(x))dx$$

$$= \frac{1}{2} \left( P_{k+1}(x) - P_k(x) \right)^c_b.$$

We also know from Churchill that $|P_n(x)| < c/n^{1/2}$, where

$$c = \left[ \frac{\pi}{2(1-x^2)} \right]^{1/2}$$

for $x$ in $(-1,1)$ and $n=1,2,...$. Thus, since this bound is relatively large, the order of convergence can be relatively small.

Now for some arbitrary initial vector $\hat{b}_0 = \hat{u}_1 + ... + \hat{u}_n$ form the sequence $\hat{b}_1, \hat{b}_2, \hat{b}_3, ...$; where $\hat{b}_1 = A\hat{b}_0$ and

$$\hat{b}_n = \frac{(2n-1)A\hat{b}_{n-1} - (n-1)\hat{b}_{n-2}}{n}$$

for $n=2,3,...,N$. We choose $N$ such that

$$\sup_{x \in (-1,1)} \left( G(x) - \sum_{n=0}^N A_n P_n(x) \right)$$

is as small as we please.

Now,

$$\hat{b}_k = P_k(\lambda_1)\hat{u}_1 + ... + P_k(\lambda_n)\hat{u}_n$$

so

$$\hat{H}(\hat{b}, c) = \sum_{k=0}^N A_k \hat{b}_k = \sum_{k=0}^N (A_k P_k(\lambda_1)\hat{u}_1 + ... + A_k P_k(\lambda_n)\hat{u}_n)$$

$$= G(\lambda_1)\hat{u}_1 + ... + G(\lambda_n)\hat{u}_n$$

$$= G(\lambda_1)\hat{u}_1 = \hat{u}_1.$$

We suggest approaching this technique from the following
point of view. The success of the above method depends on
\[ \sum_{k=0}^{N} A_k P_k(t) \] being a good approximation for \( G(t) \). Instead
of requiring such a good approximation, let us consider the
largest component, \( H_L(b,c) \), of \( \hat{H}(b,c) \). The size of this com-
ponent is due mainly to the \( i \)th vector \( G(\lambda_i) \hat{u}_i \), since
\( G(\lambda_j) = 0 \) for \( j \neq i \).

Now let us take the same vector set \( \hat{b}_0, \hat{b}_1, \hat{b}_2, \ldots \) and
consider the \( L \)th component of each. Let us also construct
a new set of constants \( A_0, A_1, A_2, \ldots \); using \( b' = (b+c)/2 \)
for the left end point instead of \( b \). Form the scalar sum,
\[ \sum_{k=0}^{N} A_k (\hat{b}_k)_L = H_L(b;c). \] If \( \lambda_i \in (b;c) \), then \( H_L(b;c) \) should
essentially equal \( H_L(b,c) \), since \( G(\lambda_i) = 1 \) and \( G(\lambda_j) = 0 \) for
\( i \neq j \). If \( \lambda_i \not\in (b,b') \) then \( H_L(b;c) \) should be considerably less
than \( H_L(b,c) \), since \( G(\lambda_i) = 0 \) for all \( i \).

Thus in either case, we have eliminated half of the
interval that \( \lambda_i \) can be in. The shift from \( b \) to \( b' \) should be
relatively fast since no matrix multiplications need be
repeated. This procedure is repeated several times to locate
\( \lambda_i \) more accurately.

This process is presented here essentially to emphasize
a different mode of approach to our general method, and, as
such, no numerical examples are given for it.
CONCLUSION

We have presented a general technique for finding the eigenvalues of a matrix whose eigenvectors span the n-dimensional complex vector space. We have illustrated three different ways of approaching this problem, all within the framework of the general method. There are certainly many more ways than we have exhibited, and possibly some of them more fruitful. We have shown that, although the amount of necessary computation is in general not insignificant, the error accumulation is not so large as to be prohibitive.

We have had the greatest success with the first technique which finds the exponential characteristic equation. This technique makes use of all accumulated information, is not limited to real eigenvalues or even to real matrices, and yields a relatively simple and accurate scheme for finding eigenvectors once the eigenvalues are known. But most importantly, this technique yields a polynomial that is well-conditioned. This means that the effect of the perturbation of the roots caused by the perturbation of a coefficient can be made as small as we please by increasing our step size h. The size of h is limited in practice, of course, due to the number of terms necessary to make the truncation error for the power series approximation to \( G(x) = e^{+x} \) insignificant. But since the terms of this power series are dominated by
Now in this technique or any similar technique as mentioned in the introduction, the polynomial coefficients are not primary data and must be calculated. Thus there is almost always some error in the computed representation of these coefficients. This technique is unique, as far as we can determine, in that it takes this error into account and reduces its effect on the roots with a proper choice of $h$ and/or a predetermined transformation of the matrix.

Therefore the accuracy of this technique is essentially determined by the accuracy of the polynomial root-finding technique. Thus we see that it is limited by the size of polynomial that the root-finding technique can solve. For example, the IBM POLRT subroutine is limited to a thirty-sixth or less degree polynomial. However, this technique can be adapted to accommodate a much larger matrix if the number of its distinct eigenvalues is within the range of the root-finder.

The matrix in example 4 has previously proved quite troublesome to the power method with deflation because of the closeness of the moduli of the roots. The technique presented here not only accurately located these roots in a relatively short time, but also illustrated the increased
well-conditioning with increasing $h$.

We believe that this technique is valuable because of its accuracy and wide applicability. But we also believe that with different choices for $G(x)$ there are techniques which are possibly as good or better. Thus we believe the general method deserves more investigation, and that this special technique deserves consideration in many practical problems.
BIBLIOGRAPHY


ACKNOWLEDGMENTS

The author sincerely wishes to thank Dr. Robert J. Lambert for his guidance in preparing this dissertation. The time and effort he spent in helping form this paper are greatly appreciated.

The author wishes also to express his gratitude to his wife, Janet T. Riess, not only for her considerable help in typing and organizing this dissertation, but for her support throughout his graduate activities.