A MATRIX SYMMETRIZATION METHOD FOR THE ALGEBRAIC EIGENPROBLEM

A thesis submitted

by

P.J. Farrell

to

the faculty of Pure and Applied Science

of the University of Ottawa

in partial fulfillment of the requirements

for the degree of

Master of Science

in the subject of

Mathematics

May 5 1964.

ACKNOWLEDGEMENTS

I am endebted to Dr. J.L. Howland who suggested this method and through whose assistance and encouragement this thesis reached completion. Also I express my sincere thanks to the National Research Council for their financial support and to my sister Mrs. Thompson who typed the thesis in its final form.

TABLE OF CONTENTS

ABSTRACT

An arbitrary matrix M is said to be symmetrized by the symmetric matrix B if the product $A = BM$ is symmetric. If such a symmetrizing matrix B is known for the $n \times n$ matrix M the eigenvalue problem $(M - rI)X = 0$ becomes $(A - rB)X = 0$ where $(A - rB)$ is symmetric.

In this thesis an extension of the Rayleigh Quotient iteration is used to solve the symmetric problem when M is real. Also in case M is non-derogatory this extension of the Rayleigh Quotient iteration is modified slightly to determine the multiplicity of multiple eigenvalues and to prevent convergence to known eigenvalues.

The method of minimized iterations is used to construct symmetrizing matrices. For when this method is applied to an arbitrary real $n \times n$ matrix M it produces a tridiagonal matrix V which is similar to M and is a direct sum of non-derogatory matrices. Also produced is a non-singular diagonal matrix D which symmetrizes V and the eigenvalue problem becomes $(DV - rD)X = 0$.

Since it is much easier to apply the method of minimized iterations to a Hessenberg matrix, orthogonal transformations are used to reduce the original matrix to Hessenberg form before the method of minimized iterations is applied.

CHAPTER 1

THE RAYLEIGH QUOTIENT ITERATION

This chapter is concerned mainly with the solution of the determinantal equation

(1.1)..... $|M - rI| = 0$

for the n eigenvalues of the n x n matrix M. The method of solution to be considered is an iterative procedure involving the use of the Rayleigh Quotient.

It is assumed that the matrix M has k distinct eigenvalues $r_1, r_2, \ldots, r_k$ and also that M is non-derogatory, i.e. if a root r_i, $1 \le i \le k$, has multiplicity m then M, considered as a linear transformation, leaves an m dimensional subspace T invariant and T is spanned by an eigenvector E_i and m-1 principal vectors $P_1, \ldots, P_{m-1}$ which satisfy the following relations:

(1.2)..... $$\begin{cases} (M - r_i I)E_i = 0 \\ (M - r_i I)P_j = P_{j-1} \quad 1 \le j \le m-1, \; P_0 = E_i. \end{cases}$$

Associated with a simple root r_i is a one dimensional invariant subspace spanned by an eigenvector E_i which satisfies the first relation in (1.2).

General Eigenvalue Problem

One method for determining the roots of M is the generalized Rayleigh Quotient iteration which is described and analysed by Ostrowski [1]. The iteration is carried out as follows:

(1.3)..... {Choose two arbitrary vectors Y_0 and X_0 such that

$Y_0 X_0 \neq 0$ and apply the recursion formulae

$$w_j = \frac{Y_j M X_j}{Y_j X_j}$$

$$(M - w_j I) X_{j+1} = X_j$$

$$Y_{j+1}(M - w_j I) = Y_j$$

successively for $j = 0, 1, 2, \ldots$.

Definition (1.4):

Let x_k and x_{k+1} denote the k'th and k+1'st terms of an iteration sequence. The sequence is said to have order of convergence p to an attractive fixed point x in case $\lim\limits_{k\to\infty} \dfrac{x_{k+1} - x}{(x_k - x)^p}$ exists and is non-zero.

The convergence of w_j in (1.3) to a root of M is shown by Ostrowski to be of third order if the root is simple and of first order if the root is multiple. There is, however, a great deal of work involved in carrying out this iteration for it requires the solution of two $n \times n$ linear systems at each step. Moreover, after convergence to a root r_1 has been obtained the iteration may converge to r_1 again during attempts to find other roots. It is shown in the next section that symmetrizing matrices are useful in overcoming the first of the above drawbacks.

The Symmetric Eigenvalue Problem

Definition (1.5):

A symmetric matrix B is a symmetrizing matrix for the matrix M if $A = BM$ is symmetric, *i.e.* $BM = M^T B$.

It is known that, for an arbitrary n x n non-derogatory matrix M, there exists an n parameter family of non-singular symmetric matrices which symmetrize M [2].

Assume now that a symmetrizing matrix B for the matrix M in (1.1) is known. Let $A = BM$. Then (1.1) can be written

(1.6)..... $|B^{-1}A - rI| = 0$

and the relations (1.2) become

$$(1.7)\ldots\ldots \begin{cases} (A - r_i B)E_i = 0 \\ (A - r_i B)F_j = BF_{j-1} \qquad 1 \leq j \leq m-1 \text{ , } F_0 = E_i \end{cases}$$

where $(A - r_i B)$ is symmetric. In this case the iteration described in (1.3) can be simplified in the following way. Choose X_0 so that $X_0^T BX_0 \neq 0$ and choose $Y_0 = X_0^T B$. Then it follows that $Y_{j+1} = X_{j+1}^T B$ for $j = 0, 1, 2, \ldots$. The formulas of (1.3) become

$$w_j = \frac{X_j^T AX_j}{X_j^T BX_j} = w_{j-1} + \frac{X_j^T BX_{j-1}}{X_j^T BX_j}$$

$$(A - w_j B)X_{j+1} = BX_j$$

and thus the solution of only one linear system is required at each step of the iteration.

Another extension of the Rayleigh - Kohn method, which is analysed by Crandall [3] and Ostrowski [4], has second order convergence to simple roots and first order convergence to multiple roots. Moreover, in the non-derogatory case it lends itself to modifications by which one can overcome the problem of reconvergence to roots previously obtained. This method will be referred to as the extended

Rayleigh - Kohn method. The method and the modifications for overcoming the problem of reconvergence are described in the next section. The discussion in the next section requires the following relations among the eigenvectors, principal vectors, and the matrices A and B.

To simplify the following it is assumed that M has k-1 simple roots $r_1, \ldots, r_{k-1}$ with associated eigenvectors $E_1, \ldots, E_{k-1}$ and a root r_k of multiplicity n-k+1 with associated eigenvector E_k and principal vectors $P_{k+1}, \ldots, P_n$. These vectors satisfy the relations

$$(1.8)\ldots\ldots \begin{cases} (A - r_jB)E_j = 0 & 1 \leq j \leq k \\ (A - r_kB)P_j = BP_{j-1} & k+1 \leq j \leq n \text{ , } P_k = E_k. \end{cases}$$

Using the symmetry of A and B, and (1.8) it follows that, for $i \neq j$ and $1 \leq i, j \leq k$,

$$E_i^TAE_j - r_jE_i^TBE_j = 0 = E_j^TAE_i - r_iE_j^TBE_i.$$

Therefore

$$(1.9)\ldots\ldots \quad E_i^TBE_j = 0 \quad \text{for} \quad i \neq j \text{ , } 1 \leq i, j \leq k.$$

Similarily

$$\left.\begin{aligned} P_j^T(A - r_kB)E_k &= 0 \\ E_k^T(A - r_kB)P_j &= E_k^TBP_{j-1} \end{aligned}\right\} \quad j = k+1, \ldots, n \text{ , } P_k = E_k$$

which yields

$$(1.10)\ldots\ldots \begin{cases} E_k^TBE_k = 0 \\ E_k^TBP_j = 0 & k+1 \leq j \leq n-1. \end{cases}$$

Also

$$\left.\begin{aligned} P_j^T(A - r_i B)E_i &= 0 \\ E_i^T(A - r_k B)P_j &= E_i^T B P_{j-1} \end{aligned}\right\} \quad 1 \le i \le k-1 \;,\; k+1 \le j \le n$$

which, upon setting $j = k+1,\ k+2,\ \ldots\ ,\ n$ in succession, yields

$$(1.11)\ldots\ldots \quad E_i^T B P_j = 0 \qquad 1 \le i \le k-1 \;,\; k+1 \le j \le n.$$

From (1.8) it follows that

$$(1.12)\ldots\ldots \quad P_i^T B P_j = P_{i-1}^T B P_{j+1} \qquad k+1 \le i \le n,\ k \le j \le n-1 \;;\; P_k = E_k$$

which together with (1.10) yields

$$(1.13)\ldots\ldots \left\{\begin{aligned} P_{k+1}^T B P_j &= 0 \qquad k \le j \le n-2 \\ P_{k+2}^T B P_j &= 0 \qquad k \le j \le n-3 \\ &\;\vdots \\ P_{k+m}^T B P_j &= 0 \qquad k \le j \le n-m-1 \\ &\;\vdots \\ P_{n-1}^T B P_j &= 0 \qquad j = k \end{aligned}\right.$$

The matrix $V = (E_1 \ldots E_k\ P_{k+1} \ldots P_n)$ is non-singular. Thus $V^T B V$ is non-singular and

$$(1.14)\ldots\ldots \left\{\begin{aligned} E_j^T B E_j &\ne 0 \qquad 1 \le j \le k-1 \\ E_k^T B P_n &\ne 0 \end{aligned}\right.$$

In the more general case where there is more than one multiple root the eigenvector and the set of principal vectors associated with each multiple root satisfy relations like (1.10) to (1.14). Moreover the invariant subspace associated with each root is B-orthogonal to the invariant subspaces associated with all the other roots, (a generalization of (1.9) and (1.11).

Extended Rayleigh - Kohn Method

This method is carried out as follows:

(1.15)..... $\begin{cases} \text{Choose a vector } R \text{ and an arbitrary number } w_0 \text{ and} \\ \text{apply the recursion relations} \\ (A - w_{j-1}B)X_j = BR \\ w_j = \dfrac{X_j^T A X_j}{X_j^T B X_j} = w_{j-1} + \dfrac{X_j^T B X_{j-1}}{X_j^T B X_j} \\ \text{successively for } j = 1, 2, 3, \ldots . \end{cases}$

The vector R is the key to preventing convergence to known roots. The system $(A - wB)X = BR$ has a unique solution X when w is not an eigenvalue. X can be written

$$X = \sum_{j=1}^{k} c_j E_j + \sum_{j=k+1}^{n} c_j P_j .$$

Also R can be written

$$R = \sum_{j=1}^{k} s_j E_j + \sum_{j=k+1}^{n} s_j P_j .$$

Using (1.9), (1.11), and (1.14) it can be shown that

(1.16)..... $c_i = s_i/(r_i - w) \qquad 1 \leq i \leq k-1$.

Also, using (1.9), (1.10), and (1.14) it follows that

(1.17)..... $c_n = s_n/(r_k - w)$.

Using (1.10), (1.11), (1.13), and (1.14) the remaining coefficients can be shown to be

(1.18)..... $\begin{cases} c_{n-1} = \dfrac{s_{n-1}}{r_k - w} - \dfrac{s_n}{(r_k - w)^2} \\ \vdots \end{cases}$

$$\begin{cases} c_{n-2} = \dfrac{s_{n-2}}{r_k-w} - \dfrac{s_{n-1}}{(r_k-w)^2} + \dfrac{s_n}{(r_k-w)^3} \\ c_k = \dfrac{s_k}{r_k-w} - \dfrac{s_{k+1}}{(r_k-w)^2} + \cdots + (-1)^{n-k}\dfrac{s_n}{(r_k-w)^{n-k+1}} \end{cases}$$

The Rayleigh Quotient $\frac{XAX}{XBX}$ is now examined in two particular cases which indicate its general properties. In the case $n=5$, $k=2$ there is a simple root r_1 with the eigenvector E_1 and a root r_2 of even multiplicity with the eigenvector E_2 and principal vectors P_3, P_4, and P_5. Let $K_{ij} = S_i^T BS_j$ where $S = E$ or P as the subscript requires. Then using (1.8) to (1.14) it can be shown that

$$XAX = r_1 c_1^2 K_{11} + r_2\left[2(c_2c_5 + c_3c_4)K_{25} + (2c_3c_5 + c_4^2)K_{35} + 2c_4c_5K_{45} + c_5^2K_{55}\right] + (2c_3c_5 + c_4^2)K_{25} + 2c_4c_5K_{35} + c_5^2K_{45}$$

$$XBX = c_1^2K_{11} + \left[2(c_2c_5 + c_3c_4)K_{25} + (2c_3c_5 + c_4^2)K_{35} + 2c_4c_5K_{45} + c_5^2K_{55}\right]$$

If the Rayleigh Quotient is put in the form

$$\frac{XAX}{XBX} = r_1 + \frac{XAX - r_1XBX}{XBX}$$

then using (1.16) it is easily seen that if $s_1 \neq 0$ and if w_{j-1} is sufficiently close to r_1, $r_1 - w_{j-1} = e$, then $w_j = r_1 + O(e^2)$ which indicates that the convergence to simple roots is of second order. If the Rayleigh Quotient is put in the form

$$\frac{XAX}{XBX} = r_2 + \frac{XAX - r_2 XBX}{XBX}$$

$$= r_2 + \frac{(r_1 - r_2)c_1^2K_{11} + (2c_3c_5 + c_4^2)K_{25} + 2c_4c_5K_{35} + c_5^2K_{45}}{c_1^2K_{11} + [2(c_2c_5 + c_3c_4)K_{25} + (2c_3c_5 + c_4^2)K_{35} + 2c_4c_5K_{45} + c_5^2K_{55}]}$$

then using (1.17) and (1.18) it follows that if $s_5 \neq 0$ and if w_{j-1} is sufficiently close to r_2, $r_2 - w_{j-1} = e$, then $w_j - r_2 \approx \frac{-3e}{4}$. If $s_5 = 0$, $s_4 \neq 0$ then the products involving c_5 disappear from the Rayleigh Quotient and $w_j - r_2 \approx \frac{-1e}{2}$. If $s_5 = s_4 = 0$ then $w_j - r_2 \approx$ (constant) in which case there is no convergence. Therefore, in general, if this method is to converge to a root r_k of even multiplicity and with principal vectors $P_{k+1}, \ldots, P_n$ then the vector R must have a component in the direction of at least one of the vectors P_j, $\frac{n+k+1}{2} \leq j \leq n$. Moreover, the convergence is of first order and the factor d (defined in Def. (1.4)) depends on the representation of the principal vectors in R.

In the case n=6, k=2 the root r_2 is of odd multiplicity. Arguments like those in the preceding case lead to the following conclusions. If $s_6 \neq 0$ then the convergence is of first order with $d = \frac{4}{5}$. If $s_6 = 0$, $s_5 \neq 0$ then the convergence is again of first order but $d = \frac{2}{3}$. If $s_6 = s_5 = 0$, $s_4 \neq 0$ then the convergence jumps to second order. If $s_6 = s_5 = s_4 = 0$ then the method does not converge to r_2. Therefore, in general, if this method is to converge to a root r_k of odd multiplicity and with principal vectors $P_{k+1}, \ldots, P_n$ then the vector R must have a component in the direction of at least one of the vectors P_j, $\frac{n+k}{2} \leq j \leq n$. The convergence is in general of first order with d depending on the representation of the principal vectors in R. The convergence is of second order if $P_{\frac{n+k}{2}}$ is represented in R and the P_j, $\frac{n+k}{2} < j \leq n$, are not represented.

To prevent convergence to known roots it is necessary to remove certain components from the vector R. If a simple root r_i is known then the component s_i in the direction of the eigenvector E_i must be removed from R. If a multiple root r_k with eigenvector E_k and principal vectors $P_{k+1}, \ldots, P_n$ is known then the components s_j in the direction of the principal vectors P_j, where $t \leq j \leq n$ and

(1.19)..... $t = \frac{n+k+1}{2}$ or $\frac{n+k}{2}$

accordingly as the multiplicity is even or odd, must be removed from R.

If E_i is the eigenvector associated with a simple root r_i then the component c_i of R in the direction of E_i is

$$c_i = \frac{E_i^T BR}{E_i^T BE_i}$$

and $c_i E_i$ can be removed from R. Similarily the components in the directions of the principal vectors $P_t, \ldots, P_n$ can be removed. For

$$c_n = \frac{P_k^T BR}{P_k^T BP_n}$$

and $c_n P_n$ can be removed from R to obtain $R_1 = R - c_n P_n$.
Then the component c_{n-1} of R_1 in the direction P_{n-1} can be removed, for

$$c_{n-1} = \frac{P_{k+1}^T BR_1}{P_{k+1}^T BP_{n-1}} .$$

Proceeding in this way the components $c_n, c_{n-1}, \ldots, c_t$ can be removed from R.

When a simple root is found using this iteration the eigenvector is also calculated and so it is a simple matter to remove from R its component in the direction of this eigenvector. If, as this iteration is carried out, the sequence w_j appears to be converging and at the same time $X_j^T BX_j$ becomes arbitrarily small then it can be concluded that w_j is converging to a multiple root. The multiplicity of the root can be determined, as in [5], by solving additional linear systems

$$(A - w_{j-1}B)V_j^{(k)} = BV_j^{(k-1)}$$

for k=1, $V_j^{(0)} = X_j$, k = 2,3, ... until the first system, for which $X_j^T BV_j^{(k)}$ does not become arbitrarily small, is found. If this occurs at k = N then the multiplicity of the root is N+1. Moreover the vectors $V_j^{(k)}$ for the final value of j are just the principal vectors. This method is based on the relations (1.7), (1.10), and (1.14). Having obtained the eigenvector and principal vectors associated with the multiple root, R can be modified to reduce the possibility of reconvergence to this multiple root.

In the following chapters a method of reducing $|M - rI| = 0$ to $|A - rB| = 0$ is described. Moreover A and B are of such a form that the system $(A - wB)X = BR$ is easily solved and the quadratic forms in the Rayleigh Quotient are easily evaluated.

CHAPTER 2

METHOD OF MINIMIZED ITERATIONS

The method of minimized iterations is a method by which an arbitrary n x n matrix can be transformed by similarity to a tridiagonal matrix. A matrix (a_{ij}) is tridiagonal if $a_{ij} = 0$ for all i and j such that $|i - j| > 1$. The elements a_{ii} comprise the main diagonal, the elements $a_{i,i+1}$ the upper codiagonal, and the elements $a_{i,i-1}$ the lower codiagonal.

The Lanczos' Algorithm [6] for carrying out the method of minimized iterations is described in this chapter. Scaling and correction procedures are also given. Cases in which the algorithm breaks down are classified and procedures for dealing with each case are detailed.

Lanczos' Algorithm

The algorithm applied to an n x n matrix A is as follows:

(2.1)..... Choose an arbitrary row vector R_1 and an arbitrary column vector C_1 (both vectors of order n) with the property $R_1 C_1 \neq 0$. Construct a sequence of row vectors $\{R_i\}_1^{n+1}$ and a sequence of column vectors $\{C_i\}_1^{n+1}$ where the vectors R_1 and C_1 are already defined and the remaining vectors are constructed from the recursion formulae

$$\left.\begin{aligned} R_{i+1} &= R_i A - \sum_{j=1}^{i} u_{ij} R_j \\ C_{i+1} &= A C_i - \sum_{j=1}^{i} v_{ji} C_j \end{aligned}\right\} 1 \le i \le n.$$

To construct R_{i+1} and C_{i+1} choose coefficients u_{ij} and v_{ji} so that

$$\left.\begin{aligned} R_{i+1} C_j &= 0 \\ R_j C_{i+1} &= 0 \end{aligned}\right\} 1 \le j \le i,$$

(i.e. so that the two sequences are biorthogonal).

It is clear that the vectors R_1 and C_1 such that $R_1 C_1 \neq 0$ can always be chosen but the question of whether the two sequences can always be constructed (i.e. the algorithm can be completed) remains. It is answered in part by the following theorem.

Theorem (2.2):

Lanczos' algorithm, applied to a matrix A, can be completed if each pair of vectors R_i and C_i constructed has the property $R_i C_i \neq 0$, $1 \le i \le n$.

Proof:

The proof is by induction. Assuming that R_1, ... , R_k and C_1, ... , C_k have been constructed and $R_i C_j = 0$, $i \neq j$, $1 \le i,j \le k$ then u_{kj} and v_{jk} , $1 \le j \le k$, defined by

$$u_{kj} = (R_k A C_j)/(R_j C_j) \text{ and } v_{jk} = (R_j A C_k)/(R_j C_j)$$

exist. Hence R_{k+1} and C_{k+1} can be constructed and

$R_{k+1} C_j = 0 = R_j C_{k+1}$ for $1 \le j \le k$.

Corollary (1):

The coefficients u_{ij} and v_{ji} are

$$\left.\begin{aligned} u_{ij} &= (R_i A C_j)/(R_j C_j) \\ v_{ji} &= (R_j A C_i)/(R_j C_j) \end{aligned}\right\} = \begin{cases} (R_i A C_i)/(R_i C_i) & 1 \le i = j \le n \\ (R_i C_i)/(R_{i-1} C_{i-1}) & 2 \le i \le n,\ j = i-1 \\ 0 & 3 \le i \le n,\ j < i-1 \end{cases}$$

Proof:

The formula $u_{ij} = (R_i A C_j)/(R_j C_j)$ is obtained by imposing the biorthogonality conditions on R_{i+1}. Therefore $u_{ii} = (R_i A C_i)/(R_i C_i)$. Also

$$\begin{aligned} u_{i,i-1} &= (R_i A C_{i-1})/(R_{i-1} C_{i-1}) \\ &= \left[R_i (C_i + \sum_{j=1}^{i-1} v_{j,i-1} C_j)\right]/(R_{i-1} C_{i-1}) \\ &= (R_i C_i)/(R_{i-1} C_{i-1}) \end{aligned}$$

because of the biorthogonality conditions previously imposed on R_i. A similar argument shows that $u_{ij} = 0$ for $j < i-1$. Similar arguments apply to v_{ji}.

Corollary (2):

The matrices $R = (R_1^T \dots R_n^T)^T$ and $C = (C_1 \dots C_n)$ are non-singular. Also $R_{n+1} = C_{n+1}^T = 0$.

Proof:

R singular implies that for some i, $1 \le i \le n$, $R_i = \sum_{\substack{j=1 \\ j \ne i}}^{n} s_j R_j$ which together with the biorthogonality conditions lead to the contradiction $R_i C_i = 0$. Similar arguments apply to 0. The biorthogonality conditions imply $R_{n+1}C = 0$ and since C is non-singular $R_{n+1} = 0$. Similar arguments yield $C_{n+1} = 0$.

<u>Corollary (3)</u>:

$RAR^{-1} = U$ a tridiagonal matrix and $C^{-1}AC = V$ a tridiagonal matrix. Also $V = U^T$.

Proof:

To illustrate this let $n = 4$. Using corollaries (1) and (2) the formulae for constructing the sequences of vectors can be rewritten

$$R_1A = R_2 + u_{11}R_1 \qquad AC_1 = C_2 + v_{11}C_1$$

$$R_2A = R_3 + u_{22}R_2 + u_{21}R_1 \qquad AC_2 = C_3 + v_{22}C_2 + v_{12}C_1$$

$$R_3A = R_4 + u_{33}R_3 + u_{32}R_2 \qquad AC_3 = C_4 + v_{33}C_3 + v_{23}C_2$$

$$R_4A = 0 + u_{44}R_4 + u_{43}R_3 \qquad AC_4 = 0 + v_{44}C_4 + v_{34}C_3$$

Therefore $RA = UR$ and $AC = CV$ where

$$U = \begin{bmatrix} u_{11} & 1 & 0 & 0 \\ u_{21} & u_{22} & 1 & 0 \\ 0 & u_{32} & u_{33} & 1 \\ 0 & 0 & u_{43} & u_{44} \end{bmatrix} \qquad V = \begin{bmatrix} v_{11} & v_{12} & 0 & 0 \\ 1 & v_{22} & v_{23} & 0 \\ 0 & 1 & v_{33} & v_{34} \\ 0 & 0 & 1 & v_{44} \end{bmatrix}$$

Since R and C are non-singular $RAR^{-1} = U$ and $C^{-1}AC = V$. From corollary (1) $u_{i,i-1} = v_{i-1,i}$ and $u_{ii} = v_{ii}$. Therefore $V = U^T$.

The problem of degeneracy, i.e. of $R_iC_i = 0$ occurring as the algorithm is being carried out, still remains and will be considered later in this chapter.

Correction Formulae

In practice it is found that if n is large then round off errors increase to the point where the biorthogonality of the two sequences is not preserved [7]. As a result some of the coefficients which were shown to be zero in corollary (1) of theorem (2.2) are in effect non-zero and the original matrix is not being transformed to a tridiagonal matrix. This problem can be overcome in the following way. For each i in turn, $2 \leq i \leq n$, compute the coefficients required to construct the vectors R_i and C_i. Assume that the biorthogonality conditions are satisfied by previously constructed vectors, then only two coefficients are computed (corollary (1) of theorem (2.2)). Next construct the vectors R_i and C_i in the usual way and then correct them by means of the following formulae:

$$(2.3)\ldots\ldots \begin{cases} R_i(\text{corrected}) = R_i - \sum_{j=1}^{i-1} \left[(R_i C_j)/(R_j C_j)\right] R_j \\ C_i(\text{corrected}) = C_i - \sum_{j=1}^{i-1} \left[(R_j C_i)/(R_j C_j)\right] C_j \end{cases}$$

In effect these formulae subtract from R_i and C_i those components which are disturbing the biorthogonality. These corrected vectors are then used to continue the algorithm.

Scaling:

In practice there is a tendency for the elements of R_i and C_i to increase in magnitude as i increases from 1 to n. If n is large it is desirable to control the magnitude of the elements. This can be done by scaling each row and column vector after it is constructed by a suitable non-zero scale factor, [8] and [9], i.e. replace the formulae in (2.1) with the following:

$$(2.4)\ldots\ldots \left\{\begin{aligned} u_{i,i+1}R_{i+1} &= R_iA - \sum_{j=1}^{i} u_{ij}R_j \\ v_{i+1,i}C_{i+1} &= AC_i - \sum_{j=1}^{i} v_{ji}C_j \end{aligned}\right\} 1 \le i \le n$$

where $u_{i,i+1}$ and $v_{i+1,i}$ are the non-zero scale factors. In this case theorem (2.2) still holds but with the following corollaries.

<u>Corollary (1'):</u>

The coefficients u_{ij} and v_{ji} are

$$u_{ij} = (R_iAC_j)/(R_jC_j) = \begin{cases} (R_iAC_i)/(R_iC_i) & 1 \le i = j \le n \\ v_{i,i-1}(R_iC_i)/(R_{i-1}C_{i-1}) & 2 \le i \le n,\ j = i-1 \\ 0 & 3 \le i \le n,\ j < i-1 \end{cases}$$

$$v_{ji} = (R_jAC_i)/(R_jC_j) = \begin{cases} (R_iAC_i)/(R_iC_i) & 1 \le i = j \le n \\ u_{i-1,i}(R_iC_i)/(R_{i-1}C_{i-1}) & 2 \le i \le n,\ j = i-1 \\ 0 & 3 \le i \le n,\ j < i-1 \end{cases}$$

<u>Corollary (2):</u>

(As before)

<u>Corollary (3')</u>

$RAR^{-1} = U$ and $C^{-1}AC = V$ where U and V are tridiagonal matrices. In the case $n = 4$

$$U = \begin{bmatrix} u_{11} & u_{12} & 0 & 0 \\ u_{21} & u_{22} & u_{23} & 0 \\ 0 & u_{32} & u_{33} & u_{34} \\ 0 & 0 & u_{43} & u_{44} \end{bmatrix} \qquad V = \begin{bmatrix} v_{11} & v_{12} & 0 & 0 \\ v_{21} & v_{22} & v_{23} & 0 \\ 0 & v_{32} & v_{33} & v_{34} \\ 0 & 0 & v_{43} & v_{44} \end{bmatrix}$$

Also if the scale factors are chosen so $u_{i,i+1} = v_{i+1,i}$ for $1 \le i \le n-1$, then $V = U^T$.

In the remainder of the thesis Lanczos' algorithm with scaling will be considered.

Breakdown of the Algorithm

The problem of degeneracy is considered in [7] and [8]. The algorithm can be carried out in the usual way until this degeneracy manifests itself through the occurrence of $B_kC_k = 0$.

Theorem (2.5):

If in applying Lanczos' algorithm to a matrix A, $B_iC_i \neq 0$ for $1 \le i \le k-1$ and $B_kC_k = 0$ then the algorithm can be carried out as far as the construction of B_k and C_k.

Corollary (1):

The coefficients u_{ij} and v_{ji}, for $1 \le i \le k-1$, are the same as those in corollary (1') of theorem (2.2) with n replaced by $k-1$.

Corollary (2):

Both sets of vectors $B_1, \dots, B_{k-1}$ and $C_1, \dots, C_{k-1}$ are linearly independent.

Corollary (3):

The formulae of the algorithm can be rewritten

$$B_1A = u_{12}B_2 + u_{11}B_1$$

$$B_2A = u_{23}B_3 + u_{22}B_2 + u_{21}B_1$$

$$\vdots$$

$$B_{k-1}A = u_{k-1,k}B_k + u_{k-1,k-1}B_{k-1} + u_{k-1,k-2}B_{k-2}$$

and

$$AC_1 = v_{21}C_2 + v_{11}C_1$$

$$AC_2 = v_{32}C_3 + v_{22}C_2 + v_{12}C_1$$

$$\vdots$$

$$AC_{k-1} = v_{k,k-1}C_k + v_{k-1,k-1}C_{k-1} + v_{k-2,k-1}C_{k-2} \text{ .}$$

The vanishing of B_kC_k prevents the continuation of the algorithm since the coefficients needed to construct B_{k+1} and C_{k+1} cannot be determined. Modifications of the algorithm, like those in [7] and [8], at the point of breakdown will now be considered. The product B_kC_k can vanish for one of the four following reasons:

$$(2.6)\ldots\ldots \begin{cases} (B1)\ldots\ldots\ B_k = 0,\ C_k \neq 0 \\ (B2)\ldots\ldots\ B_k \neq 0,\ C_k = 0 \\ (B3)\ldots\ldots\ B_k = 0,\ C_k = 0 \\ (B4)\ldots\ldots\ B_k \neq 0,\ C_k \neq 0,\ B_kC_k = 0 \end{cases}$$

Using the terminology of [10], (B1) and (B2) are unilateral breakdowns, (B3) is a bilateral breakdown, and (B4) is a deadend breakdown.

<u>Modification in Case (B1)</u>

If (B1) occurs then a vector $\bar{B}_k$ is chosen with the properties

$$(2.7)\ldots\ldots \begin{cases} \overline{R}_k C_k \neq 0 \\ \overline{R}_k C_i = 0, \quad 1 \le i \le k-1. \end{cases}$$

This is possible since

$$C_k^T - \sum_{j=1}^{k-1} \left[(C_k^T C_j)/(R_j C_j)\right] R_j$$

is such a vector. The algorithm is then carried on in the usual way with $\overline{R}_k$ replacing R_k and the following is easily proven.

<u>Theorem (2.8)</u>:

If for each pair of vectors R_i and C_i, $k+1 \le i \le n$, $R_i C_i \neq 0$ then the modified algorithm can be carried through to completion.

<u>Corollary (1)</u>:

The remaining coefficients are

$$u_{ij} = \begin{cases} (\overline{R}_k A C_k)/(\overline{R}_k C_k) & i = j = k \\ (R_i A C_i)/(R_i C_i) & k+1 \le i = j \le n \\ v_{k,k-1}(\overline{R}_k C_k)/(R_{k-1} C_{k-1}) & i = k,\ j = i-1 \\ v_{k+1,k}(R_{k+1} C_{k+1})/(\overline{R}_k C_k) & i = k+1,\ j = i-1 \\ v_{i,i-1}(R_i C_i)/(R_{i-1} C_{i-1}) & k+2 \le i \le n,\ j = i-1 \\ 0 & k \le i \le n,\ j < i-1 \end{cases}$$

and

$$v_{ji} = \begin{cases} (\overline{R}_k A C_k)/(\overline{R}_k C_k) & i = j = k \\ (R_i A C_i)/(R_i C_i) & k+1 \le i = j \le n \\ 0 & i = k,\ j = i-1 \\ u_{k,k+1}(R_{k+1} C_{k+1})/(\overline{R}_k C_k) & i = k+1,\ j = i-1 \end{cases}$$

$$\begin{cases} u_{i-1,i}(R_iC_i)/(R_{i-1}C_{i-1}) & k+2 \le i \le n,\ j = i-1 \\ 0 & k \le i \le n,\ j < i-1\ . \end{cases}$$

<u>Corollary (2):</u>

The matrices $R = (R_1^T \dots R_k^T \dots R_n^T)^T$ and $C = (C_1 \dots C_n)$ are non-singular. Also $R_{n+1} = C_{n+1}^T = 0$.

<u>Corollary (3):</u>

$RAR^{-1} = U$ and $C^{-1}AC = V$ where U and V are tri-diagonal matrices. In this case $V \neq U^T$.

Proof:

To illustrate this let $n = 4$ and $k = 3$. Then using corollary (3) of theorem (2.5) and corollaries (1) and (2) of this theorem we can write

$$\begin{aligned} R_1A &= u_{12}R_2 + u_{11}R_1 & AC_1 &= v_{21}C_2 + v_{11}C_1 \\ R_2A &= 0 + u_{22}R_2 + u_{21}R_1 & AC_2 &= v_{32}C_3 + v_{22}C_2 + v_{12}C_1 \\ R_3A &= u_{34}R_4 + u_{33}R_3 + u_{32}R_2 & AC_3 &= v_{43}C_4 + v_{33}C_3 + 0 \\ R_4A &= 0 + u_{44}R_4 + u_{43}R_3 & AC_4 &= 0 + v_{44}C_4 + v_{34}C_3 \end{aligned}$$

Therefore $RA = UR$ and $AC = CV$ where

$$U = \begin{bmatrix} u_{11} & u_{12} & 0 & 0 \\ u_{21} & u_{22} & 0 & 0 \\ 0 & u_{32} & u_{33} & u_{34} \\ 0 & 0 & u_{43} & u_{44} \end{bmatrix} \qquad V = \begin{bmatrix} v_{11} & v_{12} & 0 & 0 \\ v_{21} & v_{22} & 0 & 0 \\ 0 & v_{32} & v_{33} & v_{34} \\ 0 & 0 & v_{43} & v_{44} \end{bmatrix}$$

Since R and C are non-singular $RAR^{-1} = U$ and $C^{-1}AC = V$. $V \neq U^T$ because of the zero in the upper codiagonal of U and the non-zero scale factor v_{32} in the lower codiagonal of V.

Modification in Case (B2)

If (B2) occurs then a vector $\overline{C}_k$ is chosen with the properties

$$(2.9)\ldots\ldots \begin{cases} R_k\overline{C}_k \neq 0 \\ R_i\overline{C}_k = 0, \quad 1 \leq i \leq k-1 \end{cases}$$

This is possible since

$$R_k^T - \sum_{j=1}^{k-1} \left[(R_j R_k^T)/(R_j C_j)\right] C_j$$

is such a vector. The algorithm is then carried on in the usual way with $\overline{C}_k$ replacing C_k and it is easily proven that theorem (2.8) holds but with the following corollaries.

Corollary (1'):

The remaining coefficients are

$$u_{ij} = \begin{cases} (R_k A\overline{C}_k)/(R_k\overline{C}_k) & i = j = k \\ (R_i AC_i)/(R_i C_i) & k+1 \leq i = j \leq n \\ 0 & i = k,\ j = i-1 \\ v_{k+1,k}(R_{k+1}C_{k+1})/(R_k\overline{C}_k) & i = k+1,\ j = i-1 \\ v_{i,i-1}(R_i C_i)/(R_{i-1}C_{i-1}) & k+2 \leq i \leq n,\ j = i-1 \\ 0 & k \leq i \leq n,\ j < i-1 \end{cases}$$

and

$$v_{ji} = \begin{cases} (R_k A\overline{C}_k)/(R_k\overline{C}_k) & i = j = k \\ (R_i AC_i)/(R_i C_i) & k+1 \leq i = j \leq n \\ u_{k-1,k}(R_k\overline{C}_k)/(R_{k-1}C_{k-1}) & i = k,\ j = i-1 \\ u_{k,k+1}(R_{k+1}C_{k+1})/(R_k\overline{C}_k) & i = k+1,\ j = i-1 \\ u_{i-1,i}(R_i C_i)/(R_{i-1}C_{i-1}) & k+2 \leq i \leq n,\ j = i-1 \\ 0 & k \leq i \leq n,\ j < i-1 \end{cases}$$

Corollary (2'):

The matrices $R = (R_1^T \ldots R_n^T)^T$ and

$C = (C_1 \ldots \bar{C}_k \ldots C_n)$ are non-singular. Also $R_{n+1} = C_{n+1}^T = 0$.

Corollary (3'):

$RAR^{-1} = U$ and $C^{-1}AC = V$ where U and V are tridiagonal matrices. In this case $V \neq U^T$ and for $n = 4$, $k = 3$

$$U = \begin{bmatrix} u_{11} & u_{12} & 0 & 0 \\ u_{21} & u_{22} & u_{23} & 0 \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & u_{43} & u_{44} \end{bmatrix} \qquad V = \begin{bmatrix} v_{11} & v_{12} & 0 & 0 \\ v_{21} & v_{22} & v_{23} & 0 \\ 0 & 0 & v_{33} & v_{34} \\ 0 & 0 & v_{43} & v_{44} \end{bmatrix}$$

Modification in Case (B3)

If (B3) occurs then vectors $\bar{R}_k$ and $\bar{C}_k$ are chosen with the properties

$$(2.10)\ldots\ldots \begin{cases} \bar{R}_k\bar{C}_k \neq 0 \\ R_i\bar{C}_k = 0 \\ \bar{R}_kC_i = 0 \end{cases} \quad 1 \leq i \leq k-1$$

This is possible. For let X be any non-zero vector orthogonal to $R_1, \ldots, R_{k-1}$. Since $C_1, \ldots, C_{k-1}$ are linearly independent there exists a set of $n-k+1$ orthogonal vectors $B_k, \ldots, B_n$ such that $B_i^TC_j = 0$ for $k \leq i \leq n$, $1 \leq j \leq k-1$. The set $C_1, \ldots, C_{k-1}, B_k, \ldots, B_n$ forms a basis and

$$X = \sum_{j=1}^{k-1} s_jC_j + \sum_{j=k}^{n} s_jB_j.$$

If $XB_i = 0$ for $k \leq i \leq n$ then $X = \sum_{j=1}^{k-1} s_jC_j$ and thus

$R_i X^T \neq 0$ for some i, $1 \le i \le k-1$. Therefore $XB_i \neq 0$ for some i, $k \le i \le n$. The vector X^T is suitable as $\overline{C}_k$ and B_i^T is suitable as $\overline{R}_k$. In practice the vectors can be chosen by a procedure of trial and error [7].

After choosing $\overline{R}_k$ and $\overline{C}_k$ the algorithm is carried on in the usual way with $\overline{R}_k$ and $\overline{C}_k$ replacing R_k and C_k. It is easily proven that again theorem (2.8) holds but with the following corollaries.

<u>Corollary (1''):</u>

The remaining coefficients are

$$u_{ij} = \begin{cases} (\overline{R}_k A\overline{C}_k)/(\overline{R}_k\overline{C}_k) & i = j = k \\ (R_i AC_i)/(R_i C_i) & k+1 \le i = j \le n \\ 0 & i = k,\ j = i-1 \\ v_{k+1,k}(R_{k+1}C_{k+1})/(\overline{R}_k\overline{C}_k) & i = k+1,\ j = i-1 \\ v_{i,i-1}(R_i C_i)/(R_{i-1}C_{i-1}) & k+2 \le i \le n,\ \ j = i-1 \\ 0 & k \le i \le n,\ \ j < i-1 \end{cases}$$

and

$$v_{ji} = \begin{cases} (\overline{R}_k A\overline{C}_k)/(\overline{R}_k\overline{C}_k) & i = j = k \\ (R_i AC_i)/(R_i C_i) & k+1 \le i = j \le n \\ 0 & i = k,\ \ j = i-1 \\ u_{k,k+1}(R_{k+1}C_{k+1})/(\overline{R}_k\overline{C}_k) & i = k+1,\ \ j = i-1 \\ u_{i-1,i}(R_i C_i)/(R_{i-1}C_{i-1}) & k+2 \le i \le n,\ \ j = i-1 \\ 0 & k \le i \le n,\ j < i-1 \end{cases}$$

Corollary (2''):

The matrices $R = (R_1^T \ldots \overline{R}_k^T \ldots R_n^T)^T$ and $C = (C_1 \ldots \overline{C}_k \ldots C_n)$ are non-singular. Also $R_{n+1} = C_{n+1}^T = 0$.

Corollary (3''):

$RAR^{-1} = U$ and $C^{-1}AC = V$ where U and V are tridiagonal matrices. In this case $V = U^T$ if the scale factors are chosen so $u_{i-1,i} = v_{i,i-1}$ and for $n = 4$, $k = 3$

$$U = \begin{bmatrix} u_{11} & u_{12} & 0 & 0 \\ u_{21} & u_{22} & 0 & 0 \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & u_{43} & u_{44} \end{bmatrix} \qquad V = \begin{bmatrix} v_{11} & v_{12} & 0 & 0 \\ v_{21} & v_{22} & 0 & 0 \\ 0 & 0 & v_{33} & v_{34} \\ 0 & 0 & v_{43} & v_{44} \end{bmatrix}$$

Breakdowns of type (B3) result in tridiagonal matrices U and V which are direct sums of matrices. This is advantageous in finding eigenvalues for it reduces the problem to that of finding the eigenvalues of matrices which are of smaller dimension than the original.

Multiple Breakdowns

Up to now only a single occurrence of one of (B1), (B2) or (B3) has been considered. In each of the three cases the possibility of modifying the algorithm does not depend on the non-occurrence of (B1), (B2) or (B3) at some preceding stage. Thus in carrying out the algorithm unilateral and bilateral breakdowns may appear in any order and with any frequency and the algorithm with appropriate modifications can be completed, provided (B4) does not occur. Moreover, a pair of tridiagonal matrices both similar to A can be constructed. The coefficients on the right hand side of the formulae (2.4) are given by corollary (1') of theorem (2.2) from the beginning of the algorithm until the first breakdown. After a breakdown of type (B1), (B2) or (B3) the coefficients are given by corollary (1), (1') or (1'') respectively, of theorem (2.8), but only as far as the next breakdown.

If no breakdowns or bilateral breakdowns only occur then the pair of tridiagonal matrices constructed are connected by the property that each is the transpose of the other, provided of course that the scale factors $u_{i,i+1}$ and $v_{i+1,i}$ are chosen equal throughout the algorithm.

Modification in Case (B4)

If (B4) occurs there appears to be no alternative but to restart the algorithm with different starting vectors B_1 and C_1 or restart the algorithm at a point where a breakdown of type (B1), (B2) or (B3) has occurred. That there exist starting vectors for which (B4) will never occur is shown in 8 . In practice, however, the usual procedure is to choose arbitrary starting vectors. Thus the possibility of a deadend breakdown occurring remains. If it does occur the algorithm must be restarted.

Causes of Degeneracy

Methods of continuing the algorithm in degenerate cases have been considered. Some of the causes of degeneracy are now described.

<u>Theorem (2.11):</u>

If the vectors $R_1, R_1A, \ldots, R_1A^k$, $k < n$, are linearly dependent then (B1) or (B3) must occur. If the vectors $C_1, AC_1, \ldots, A^kC_1$, $k < n$, are linearly dependent then (B2) or (B3) must occur.

Proof:

Suppose R_{k+1} has been constructed without (B1) or (B3) occurring. It is easily seen from the recursion formulae (2.4) that R_{k+1} is a linear combination of $R_1, \ldots R_1A^k$. Therefore $R_1, \ldots, R_{k+1}$ are linearly dependent. But $R_i = \sum_{\substack{j=1 \\ j\neq i}}^{k+1} s_jR_j$ for any i, $1 \leq i \leq k$, leads to the contradiction $R_iC_i = 0$. Therefore $R_{k+1} = \sum_{j=1}^{k} s_jR_j$. $R_{k+1} \neq 0$ leads to the contradiction $R_{k+1}C_i \neq 0$ for at least one i, $1 \leq i \leq k$. Therefore $R_{k+1} = 0$ i.e. (B1) or (B3) occurs.

A similar argument applies to the C vectors.

<u>Corollary:</u>

If the matrix A is derogatory then (B1) and (B2), or (B3) must occur.

Proof:

A being derogatory, the degree m of the minimal polynomial of A is less than n the order of A. Let M(x) be the minimal polynomial. Then $M(A) = 0$ and for any vectors R_1 and C_1 $R_1M(A) = 0 = M(A)C_1$. Hence the conditions of the theorem are satisfied with $k = n$.

The preceding theorem and corollary indicate that degeneracy depends on the matrix A and on the starting vectors B_1 and C_1.

CHAPTER 3

CALCULATION OF SYMMETRIZING MATRICES

Symmetrizing matrices were defined in definition (1.5) of chapter 1. It will now be shown that the method of minimized iterations described in chapter 2 can be used to construct such symmetrizing matrices.

Construction of a Symmetrizing Matrix

The method of minimized iterations when applied to a matrix A produces a pair of tridiagonal matrices U and V both similar to A. Also produced are a sequence of n row vectors $\{R_i\}$ and a sequence of n column vectors $\{C_i\}$. The two sequences are biorthogonal i.e. $R_iC_j = 0$ for $i \neq j$. They also have the property $R_iC_i \neq 0$. Thus $RC = D$ a non-singular diagonal matrix, where $R = (R_1^T \dots R_n^T)^T$ and $C = (C_1 \dots C_n)$. Moreover $RA = UR$ and $AC = CV$. Therefore (3.1)..... $RAC = UD = DV$

If in carrying out the algorithm of the method of minimized iterations no breakdowns occur then $V = U^T$, (corollary (3') of theorem (2.2)), in which case (3.1) yields $DV = V^TD$. Thus the method of minimized iterations produces a tridiagonal matrix V and a diagonal matrix D which symmetrizes V. In terms of the elements of V and D the matrices V, D, and DV are, in the case $n=4$ (theorem (2.2))

$$V = \begin{bmatrix} \frac{R_1AC_1}{R_1C_1} & \frac{u_{12}R_2C_2}{R_1C_1} & 0 & 0 \\ v_{21} & \frac{R_2AC_2}{R_2C_2} & \frac{u_{23}R_3C_3}{R_2C_2} & 0 \\ 0 & v_{32} & \frac{R_3AC_3}{R_3C_3} & \frac{u_{34}R_4C_4}{R_3C_3} \\ 0 & 0 & v_{43} & \frac{R_4AC_4}{R_4C_4} \end{bmatrix}$$

$$D = \begin{bmatrix} R_1C_1 & & & \\ & R_2C_2 & & \\ & & R_3C_3 & \\ & & & R_4C_4 \end{bmatrix}$$

$$DV = \begin{bmatrix} R_1AC_1 & u_{12}R_2C_2 & 0 & 0 \\ v_{21}R_2C_2 & R_2AC_2 & u_{23}R_3C_3 & 0 \\ 0 & v_{32}R_3C_3 & R_3AC_3 & u_{34}R_4C_4 \\ 0 & 0 & v_{43}R_4C_4 & R_4AC_4 \end{bmatrix}$$

which is symmetric when the scale factors $u_{i,i+1}$ and $v_{i+1,i}$ are chosen equal.

If in carrying out the algorithm breakdowns of type (B3) only occur then $V = U^T$, (corollary (3'') of theorem (2.8)), in which case (3.1) yields $DV = V^TD$, i.e. D symmetrizes V. In terms of the elements of V and D the matrices V, D, and DV are in the case $n = 4$ (theorems (2.5) and (2.8) with corollaries (1''), (2''), and (3''))

$$V = \begin{bmatrix} \frac{R_1AC_1}{R_1C_1} & \frac{u_{12}R_2C_2}{R_1C_1} & 0 & 0 \\ v_{21} & \frac{R_2AC_2}{R_2C_2} & 0 & 0 \\ 0 & 0 & \frac{\bar{R}_3A\bar{C}_3}{\bar{R}_3\bar{C}_3} & \frac{u_{34}R_4C_4}{\bar{R}_3\bar{C}_3} \\ 0 & 0 & v_{43} & \frac{R_4AC_4}{R_4C_4} \end{bmatrix}$$

$$D = \begin{bmatrix} R_1C_1 & & & \\ & R_2C_2 & & \\ & & \bar{R}_3\bar{C}_3 & \\ & & & R_4C_4 \end{bmatrix}$$

$$DV = \begin{bmatrix} R_1AC_1 & u_{12}R_2C_2 & 0 & 0 \\ v_{21}R_2C_2 & R_2AC_2 & 0 & 0 \\ 0 & 0 & \bar{R}_3A\bar{C}_3 & u_{34}R_4C_4 \\ 0 & 0 & v_{43}R_4C_4 & R_4AC_4 \end{bmatrix}$$

which is symmetric when the scale factors $u_{i,i+1}$ and $v_{i+1,i}$ are equal.

As was mentioned in chapter 2 breakdowns of type (B3) result in a reduction of the problem to that of finding the roots of several smaller matrices.

If in carrying out the algorithm a breakdown of type (B1) occurs then $V \neq U^T$, (corollary (3) of theorem (2.8)). However in terms of the elements of V and D the matrices V, D, and DV are in the case n = 4 (theorems (2.5) and

(2.8) with corollaries (1), (2), and (3))

$$V = \begin{bmatrix} \dfrac{R_1AC_1}{R_1C_1} & u_{12}\dfrac{R_2C_2}{R_1C_1} & 0 & 0 \\ v_{21} & \dfrac{R_2AC_2}{R_2C_2} & 0 & 0 \\ 0 & v_{32} & \dfrac{\bar{R}_3AC_3}{\bar{R}_3C_3} & u_{34}\dfrac{R_4C_4}{\bar{R}_3C_3} \\ 0 & 0 & v_{43} & \dfrac{R_4AC_4}{R_4C_4} \end{bmatrix}$$

$$D = \begin{bmatrix} R_1C_1 & & & \\ & R_2C_2 & & \\ & & \bar{R}_3C_3 & \\ & & & R_4C_4 \end{bmatrix}$$

$$DV = \begin{bmatrix} R_1AC_1 & u_{12}R_2C_2 & 0 & 0 \\ v_{21}R_2C_2 & R_2AC_2 & 0 & 0 \\ 0 & v_{32}\bar{R}_3C_3 & \bar{R}_3AC_3 & u_{34}R_4C_4 \\ 0 & 0 & v_{43}R_4C_4 & R_4AC_4 \end{bmatrix}$$

which would be symmetric if $(DV)_{23}$ were equal to $(DV)_{32}$. The symmetry is spoiled by a zero element in the upper codiagonal. If k breakdowns of type (B1) occur then the symmetry of DV is spoiled by k zero elements in the upper codiagonal.

If in carrying out the algorithm a breakdown of type (B2) occurs then $V \neq U^T$, (corollary (3') of theorem (2.8). However, in terms of the elements of V and D, the matrices V, D, and DV are in the case $n = 4$ (theorems (2.5) and (2.8) with corollaries (1'), (2'), and (3'))

$$V = \begin{bmatrix} \dfrac{R_1AC_1}{R_1C_1} & \dfrac{u_{12}R_2C_2}{R_1C_1} & 0 & 0 \\[2ex] v_{21} & \dfrac{R_2AC_2}{R_2C_2} & \dfrac{u_{23}R_3\bar{C}_3}{R_2C_2} & 0 \\[2ex] 0 & 0 & \dfrac{R_3A\bar{C}_3}{R_3\bar{C}_3} & \dfrac{u_{34}R_4C_4}{R_3\bar{C}_3} \\[2ex] 0 & 0 & v_{43} & \dfrac{R_4AC_4}{R_4C_4} \end{bmatrix}$$

$$D = \begin{bmatrix} R_1C_1 & & & \\ & R_2C_2 & & \\ & & R_3\bar{C}_3 & \\ & & & R_4C_4 \end{bmatrix}$$

$$DV = \begin{bmatrix} R_1AC_1 & u_{12}R_2C_2 & 0 & 0 \\ v_{21}R_2C_2 & R_2AC_2 & u_{23}R_3\bar{C}_3 & 0 \\ 0 & 0 & R_3A\bar{C}_3 & u_{34}R_4C_4 \\ 0 & 0 & v_{43}R_4C_4 & R_4AC_4 \end{bmatrix}$$

which would be symmetric if $(DV)_{23}$ were equal to $(DV)_{32}$. The symmetry is spoiled by a zero element in the lower co-diagonal. If k breakdowns of type (B2) occur then the symmetry of DV is spoiled by k zero elements in the lower codiagonal.

It is interesting to note also that, since $RA = UR$ and $AC = CV$, in those cases in which $V = U^T$

$$A^T R^T = R^T U^T = R^T(C^{-1}AC).$$

Therefore

$$(R^T C^{-1})A = A^T(R^T C^{-1}).$$

Also since $RC = D$, a diagonal matrix,

$$(R^T C^{-1})^T = (C^{-1})^T R = (D^{-1}R)^T DC^{-1} = R^T C^{-1}.$$

Therefore $R^T C^{-1}$ symmetrizes the matrix A.

It has been shown that the method of minimised iterations produces a matrix DV which in some cases is symmetric and in others is almost symmetric. Both the symmetric and almost symmetric matrices will prove useful in the solution of the eigenvalue problem. For this reason it is assumed in the next chapters that the scale factors $u_{i,i+1}$ and $v_{i+1,i}$ are always chosen equal.

CHAPTER 4

ADVANTAGES OF HESSENBERG FORM

Hessenberg Matrices

A $n \times n$ matrix A is upper (lower) Hessenberg if $a_{ij} = 0$ for all i and j such that $j < i-1$ $(j > i+1)$. A 4×4 upper Hessenberg matrix A has the form

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ 0 & a_{32} & a_{33} & a_{34} \\ 0 & 0 & a_{43} & a_{44} \end{bmatrix}$$

This chapter is concerned with the simplifications which occur in Lanczos' algorithm when the matrix A is upper Hessenberg. Unilateral and bilateral breakdowns and practical methods for continuing the algorithm when they occur are again considered.

Lanczos' Algorithm

The work involved in applying Lanczos' algorithm to a $n \times n$ Hessenberg matrix is far less than for a general $n \times n$ matrix. The reasons for this are mentioned in [11]. They are included in the following theorem and corollaries.

Theorem (4.1):

Let A be a $n \times n$ upper Hessenberg matrix. Let $R_1 = (r_{11}, \ldots, r_{1n})$ and $C_1 = (c_{11}, 0, \ldots, 0)^T$ with $r_{11} \neq 0$ and $c_{11} \neq 0$ be the starting vectors in

Lanczos' algorithm. If for each pair of vectors R_i and C_i constructed $R_iC_i \neq 0$, $1 \leq i \leq n$, then the matrices

$$R = (R_1^T \cdots R_n^T)^T \quad \text{and} \quad C = (C_1 \cdots C_n)$$

are both upper triangular.

Proof:

Let $R_i = (r_{i1}, \cdots, r_{in})$ and $C_i = (c_{1i}, \cdots, c_{ni})^T$ for each i, $1 \leq i \leq n$. Then $c_{j1} = 0$ for $2 \leq j \leq n$.

Theorem (2.2) with corollaries (1'), (2), and (3') holds in this case and ensures that the algorithm can be completed. The form of R and C must still be checked. Due to the special form of A and C_1

$$R_1AC_1 = r_{11}(a_{11}c_{11}) + r_{12}(a_{21}c_{11})$$

$$R_1C_1 = r_{11}c_{11} \neq 0.$$

Therefore

$$u_{11} = v_{11} = \frac{R_1AC_1}{R_1C_1} = a_{11} + \frac{r_{12}}{r_{11}}a_{21}.$$

$$u_{12}R_2 = R_1A - u_{11}R_1 \quad \text{and} \quad v_{21}C_2 = AC_1 - v_{11}C_1$$

from which it follows

$$u_{12}r_{21} = (r_{11}a_{11} + r_{12}a_{21}) - u_{11}r_{11} = 0$$

$$v_{21}c_{12} = a_{11}c_{11} - v_{11}c_{11} = \frac{-r_{12}a_{21}c_{11}}{r_{11}}$$

$$v_{21}c_{22} = a_{21}c_{11}$$

$$v_{21}c_{j2} = 0 \quad \text{for} \quad j > 2.$$

Therefore

$$\frac{c_{12}}{c_{22}} = \frac{-r_{12}}{r_{11}}$$

$$R_2 = (0, r_{22}, \dots, r_{2n}) \text{ and } C_2 = (c_{12}, c_{22}, 0, \dots, 0)^T.$$

Due to the form of A and the vectors R_2 and C_2

$$R_2AC_2 = r_{22}(a_{21}c_{12} + a_{22}c_{22}) + r_{23}(a_{32}c_{22})$$

$$R_2C_2 = r_{22}c_{22} \neq 0.$$

Therefore, since the scale factors u_{12} and v_{21} are chosen equal,

$$u_{21} = v_{21}\frac{R_2C_2}{R_1C_1} = v_{21}\frac{r_{22}c_{22}}{r_{11}c_{11}} = \frac{r_{22}}{r_{11}}a_{21}$$

$$v_{12} = u_{12}\frac{R_2C_2}{R_1C_1} = u_{21}$$

$$u_{22} = v_{22} = \frac{R_2AC_2}{R_2C_2} = a_{21}\frac{c_{12}}{c_{22}} + a_{22} + \frac{r_{23}}{r_{22}}a_{32}$$

$$= \frac{-r_{12}}{r_{11}}a_{21} + a_{22} + \frac{r_{23}}{r_{22}}a_{32}.$$

By an argument using induction it can be shown that

$$\left.\begin{aligned} &(4.2)\dots\dots \quad R_i = (0, \dots, 0, r_{ii}, \dots, r_{in}) \\ &(4.3)\dots\dots \quad C_i = (c_{1i}, \dots, c_{ii}, 0, \dots, 0)^T \end{aligned}\right\} 1 \le i \le n$$

$$\left.\begin{aligned} &(4.4)\dots\dots \quad v_{i,i-1}c_{ii} = a_{i,i-1}c_{i-1,i-1} \\ &(4.5)\dots\dots \quad \frac{c_{i-1,i}}{c_{ii}} = \frac{-r_{i-1,i}}{r_{i-1,i-1}} \end{aligned}\right\} 2 \le i \le n$$

$$(4.6)\dots\dots \quad R_iAC_i = \begin{cases} r_{11}(a_{11}c_{11}) + r_{12}a_{21}c_{11} & i = 1 \\ r_{ii}(a_{i,i-1}c_{i-1,i} + a_{ii}c_{ii}) + r_{i,i+1}a_{i+1,i}c_{ii} & 2 \le i \le n-1 \end{cases}$$

$$\begin{cases} r_{nn}(a_{n,n-1}c_{n-1,n} + a_{nn}c_{nn}) & i = n \end{cases}$$

$$(4.7)\ldots\ldots \quad R_iC_i = r_{ii}c_{ii} \quad 1 \le i \le n$$

$$\left.\begin{aligned} &(4.8)\ldots\ldots \quad u_{i,i-1} = v_{i,i-1}\frac{R_iC_i}{R_{i-1}C_{i-1}} = \frac{r_{ii}a_{i,i-1}}{r_{i-1,i-1}} \\ &(4.9)\ldots\ldots \quad v_{i-1,i} = u_{i-1,i}\frac{R_iC_i}{R_{i-1}C_{i-1}} = u_{i,i-1} \end{aligned}\right\} 2 \le i \le n$$

$$(4.10)\ldots\ldots \quad u_{ii} = v_{ii} = \frac{R_iAC_i}{R_iC_i}$$

$$= \begin{cases} a_{11} + \dfrac{r_{12}a_{21}}{r_{11}} & i = 1 \\ \dfrac{c_{i-1,i}a_{i,i-1}}{c_{ii}} + a_{ii} + \dfrac{r_{i,i+1}a_{i+1,i}}{r_{ii}} & 2 \le i \le n-1 \\ \dfrac{c_{n-1,n}a_{n,n-1}}{c_{nn}} + a_{nn} & i = n. \end{cases}$$

<u>Corollary (1):</u>

The sequence of row vectors $\{R_i\}$ and the tri-diagonal matrices U and V can be constructed without constructing the sequence of column vectors $\{C_i\}$.

Proof:

From (4.4)

$$(4.11)\ldots\ldots \quad c_{ii} = \prod_{j=2}^{i} \frac{a_{j,j-1}c_{11}}{v_{j,j-1}} = \prod_{j=2}^{i} \frac{a_{j,j-1}c_{11}}{u_{j-1,j}} \quad 2 \le i \le n.$$

(4.11) together with (4.5), (4.8), and (4.10) show that the coefficients u_{ii} and $u_{i,i-1}$ can be computed from the preceding R vectors and the elements of A.

Corollary (2):

The diagonal matrix $D = RC$ and the symmetric tridiagonal matrix DV can be constructed without constructing the sequence of column vectors $\{C_i\}$.

Proof:

It was shown in chapter 3 that the elements of D and DV are R_iC_i, R_iAC_i, and $u_{i-1,i}R_iC_i$. (4.7) and (4.11) show that the C_i vectors are not required for R_iC_i. (4.5), (4.6), and (4.11) show the same for R_iAC_i.

Corollary (3):

The correction formulae (2.3) of chapter 2 are not required.

Proof:

For each i, the vector R_i is of the form $(0, \ldots, 0, r_{ii}, \ldots, r_{in})$. In practice the first i-1 components of R_i are not computed but set to zero. This ensures that R_i is orthogonal to all the preceding C_j vectors since they are of the form $(c_{1j}, \ldots, c_{jj}, 0, \ldots, 0)^T$. If the sequence of column vectors $\{C_i\}$ is not calculated then their correction formulae are not needed.

In practice the algorithm is carried out in the following way:

Choose R_1 and C_1.

Compute R_1C_1 and R_1AC_1 from (4.7) and (4.6).

Compute u_{11} from $u_{11} = \dfrac{R_1AC_1}{R_1C_1}$.

(*)..... Construct $u_{12}R_2$, choose the scale factor u_{12}, and compute R_2.

Compute c_{22} from (4.11).

Compute R_2C_2, $u_{12}R_2C_2$, and R_2AC_2 from (4.7), (4.5), and (4.6).

Construct $u_{21} = \dfrac{u_{12}R_2C_2}{R_1C_1}$ and $u_{22} = \dfrac{R_2AC_2}{R_2C_2}$.

Repeat the procedure from (*) for $u_{23}R_3$, $u_{34}R_4$, etc.

The elements of the symmetric tridiagonal and diagonal matrices are obtained in the preceding scheme.

<u>Breakdown of the Algorithm</u>

Breakdown of the algorithm as applied to a general matrix A was considered in chapter 2. It is now reconsidered for an upper Hessenberg matrix A.

<u>Theorem(4.12)</u>:

Let A, R_1, and C_1 be as in theorem (4.1). If $R_iC_i \neq 0$ for $1 \leq i \leq k-1$ and $R_kC_k = 0$ then for $1 \leq i \leq k$ the vectors R_i and C_i are of the form

$$R_i = (0, \ldots, 0, r_{ii}, \ldots, r_{in}) \text{ and}$$

$$C_i = (c_{1i}, \ldots, c_{ii}, 0, \ldots, 0)^T.$$

Proof:

Theorem (2.5) and its corollaries hold in this case and the proof proceeds as in theorem (4.1). In carrying out the proof one obtains:

(4.2), (4.3), (4.4), and (4.7) with k replacing n.

(4.5), (4.8), and (4.9) with k-1 replacing n.

The first two parts of (4.6) and (4.10) with k-1 replacing n-1.

<u>Corollary (1)</u>:

The sequence of row vectors $R_1, \ldots, R_k$ can be constructed without constructing the sequence of column vectors $C_1, \ldots, C_k$.

In chapter 2 it was shown that the breakdown $R_kC_k = 0$ can be attributed to one of four causes given in (2.6). When A is upper Hessenberg and the starting vectors are chosen as indicated it follows from theorem (4.12) that $R_kC_k = r_{kk}c_{kk}$. Therefore $R_kC_k = 0$ if and only if $r_{kk} = 0$ or $c_{kk} = 0$. The vector C_k, however, has the following properties.

<u>Theorem (4.13)</u>:

$C_k = 0$ if and only if $c_{kk} = 0$ if and only if $a_{k,k-1} = 0$.

Proof:

From (4.4) it follows that

$$c_{kk} = \frac{a_{k,k-1}c_{k-1,k-1}}{v_{k,k-1}}.$$

But $R_{k-1}C_{k-1} = r_{k-1,k-1}c_{k-1,k-1} \neq 0$ and $v_{k,k-1}$ is a non-zero scale factor. Therefore $c_{kk} = 0$ if and only if $a_{k,k-1} = 0$.

Clearly $C_k = 0$ implies $c_{kk} = 0$. $R_i C_i = r_{ii} c_{ii} \neq 0$ which implies $r_{ii} \neq 0$ for $1 \leq i \leq k-1$. Therefore, if $c_{kk} = 0$, $R_{k-1} C_k = r_{k-1,k-1} c_{k-1,k} = 0$ which implies $c_{k-1,k} = 0$. By repeating the argument for $k-2, \ldots, 1$ it follows that $C_k = 0$.

The four possible causes of the breakdown $R_k C_k = 0$ can now be rewritten

$$(4.14)\ldots\ldots \begin{cases} (B1)\ldots\ldots & R_k = 0, \quad a_{k,k-1} \neq 0 \quad \text{(unilateral breakdown)} \\ (B2)\ldots\ldots & r_{kk} \neq 0, \quad a_{k,k-1} = 0 \quad \text{(unilateral breakdown)} \\ (B3)\ldots\ldots & R_k = 0, \quad a_{k,k-1} = 0 \ \text{(bilateral breakdown)} \\ (B4)\ldots\ldots & R_k \neq 0, \ r_{kk} = 0 \quad \text{(deadend breakdown)} \end{cases}$$

<u>Modification in Case (B1)</u>

As described in chapter 2 a vector $\overline{R}_k$ with property (2.7) is chosen. In this case any vector of the form $(0, \ldots, 0, x_k, \ldots, x_n)$, $x_k \neq 0$, has this property. Thus the vector $\overline{R}_k$ is easily chosen and the algorithm continued.

<u>Theorem (4.15)</u>:

If for each pair of vectors R_i and C_i, $k+1 \leq i \leq n$, $R_i C_i \neq 0$ then the matrices $R = (R_1^T \ldots \overline{R}_k^T \ldots R_n^T)^T$ and $C = (C_1 \ldots C_n)$ are both upper triangular.

Proof:

Theorem (2.8) with corollaries (1), (2), and (3) holds in this case. The proof proceeds as in theorem

(4.1) and the following formulae are obtained:

$$\left.\begin{aligned} R_i &= (0, \dots, 0, r_{ii}, \dots, r_{in}) \\ C_i &= (c_{1i}, \dots, c_{ii}, 0, \dots, 0)^T \\ v_{i,i-1}c_{ii} &= a_{i,i-1}c_{i-1,i-1} \end{aligned}\right\} \quad k+1 \le i \le n$$

$$\frac{c_{i-1,i}}{c_{ii}} = \begin{cases} \dfrac{-r_{i-1,i}}{r_{i-1,i-1}} & k \le i \le n \quad i \ne k+1 \\ \dfrac{-\bar{r}_{k,k+1}}{\bar{r}_{kk}} & i = k+1 \end{cases}$$

$$\bar{R}_k A C_k = \bar{r}_{kk}(a_{k,k-1}c_{k-1,k} + a_{kk}c_{kk}) + \bar{r}_{k,k+1}a_{k+1,k}c_{kk}$$

$$R_i A C_i = \begin{cases} r_{ii}(a_{i,i-1}c_{i-1,i} + a_{ii}c_{ii}) + r_{i,i+1}a_{i+1,i}c_{ii} & k+1 \le i \le n-1 \\ r_{nn}(a_{n,n-1}c_{n-1,n} + a_{nn}c_{nn}) & i = n \end{cases}$$

$$\bar{R}_k C_k = \bar{r}_{kk}c_{kk}$$

$$R_i C_i = r_{ii}c_{ii} \qquad k+1 \le i \le n$$

$$u_{i,i-1} = \begin{cases} \dfrac{\bar{r}_{kk}a_{k,k-1}}{r_{k-1,k-1}} & i = k \\ \dfrac{r_{k+1,k+1}a_{k+1,k}}{\bar{r}_{kk}} & i = k+1 \\ \dfrac{r_{ii}a_{i,i-1}}{r_{i-1,i-1}} & k+2 \le i \le n \end{cases}$$

$$v_{i-1,i} = \begin{cases} 0 & i = k \\ u_{i,i-1} & k+1 \le i \le n \end{cases}$$

$$u_{ii} = v_{ii} = \frac{c_{i-1,i}}{c_{ii}}a_{i,i-1} + a_{ii} + \frac{r_{i,i+1}}{r_{ii}}c_{i+1,i} \qquad k \le i \le n-1$$

$$u_{nn} = v_{nn} = \frac{c_{n-1,n}a_{n,n-1}}{c_{nn}} + a_{nn}.$$

Corollary (1):

The sequence of row vectors $\bar{R}_1, \ldots, \bar{R}_k, \ldots, \bar{R}_n$ and the tridiagonal matrices U and V can be constructed without constructing the sequence of column vectors $\{\bar{C}_i\}$.

Proof:

This follows from the formulae of this theorem and theorem (4.12). Again it follows that

$$c_{ii} = \prod_{j=2}^{i} \frac{a_{j,j-1}}{v_{j,j-1}} c_{11}.$$

Corollary (2):

The diagonal matrix $D = RC$ and the almost symmetric tridiagonal matrix DV can be constructed without constructing the sequence of column vectors $\{\bar{C}_i\}$.

Modification in Case (B2)

As described in chapter 2 a vector $\bar{C}_k$ with property (2.9) is chosen. The linear system

$$(4.16)\ldots\ldots \quad \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1,k-1} & r_{1k} \\ & r_{22} & \cdots & r_{2,k-1} & r_{2k} \\ & 0 & \ddots & \vdots & \vdots \\ & & & r_{k-1,k-1} & r_{k-1,k} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \end{bmatrix} = 0$$

has a solution for any non-zero x_k and if $(x_1, \ldots, x_k)^T$, $x_k \neq 0$, denotes such a solution then the vector

$(x_1, \ldots, x_k, 0, \ldots, 0)^T$ may be chosen as $\bar{C}_k$ and the algorithm continued.

Theorem (4.17):

If for each pair of vectors R_i and C_i, $k+1 \leq i \leq n$, $R_i C_i \neq 0$ then the matrices

$$R = (R_1^T \ldots R_n^T)^T \quad \text{and} \quad C = (C_1 \ldots \bar{C}_k \ldots C_n)$$

are both upper triangular.

Proof:

Theorem (2.8) with corollaries (1'), (2'), and (3') holds in this case. The proof proceeds like in theorem (4.1) and the following formulae are obtained:

$$\left.\begin{aligned} R_i &= (0, \ldots, 0, r_{ii}, \ldots, r_{in}) \\ C_i &= (c_{1i}, \ldots, c_{ii}, 0, \ldots, 0)^T \end{aligned}\right\} \quad k+1 \leq i \leq n$$

$$v_{k+1,k} c_{k+1,k+1} = a_{k+1,k} \bar{c}_{kk}$$

$$v_{i,i-1} c_{ii} = a_{i,i-1} c_{i-1,i-1} \qquad k+2 \leq i \leq n$$

$$\frac{\bar{c}_{k-1,k}}{\bar{c}_{kk}} = \frac{-r_{k-1,k}}{r_{k-1,k-1}}$$

$$\frac{c_{i-1,i}}{c_{ii}} = \frac{-r_{i-1,i}}{r_{i-1,i-1}} \qquad k+1 \leq i \leq n$$

$$R_k A \bar{C}_k = r_{kk} a_{kk} \bar{c}_{kk} + r_{k,k+1} a_{k+1,k} \bar{c}_{kk}$$

$$R_i A C_i = \begin{cases} r_{ii}(a_{i,i-1} c_{i-1,i} + a_{ii} c_{ii}) + r_{i,i+1} a_{i+1,i} c_{ii} & k+1 \leq i \leq n-1 \\ r_{nn}(a_{n,n-1} c_{n-1,n} + a_{nn} c_{nn}) & i = n \end{cases}$$

$$R_k \bar{C}_k = r_{kk} \bar{c}_{kk}$$

$$R_i C_i = r_{ii} c_{ii} \qquad k+1 \le i \le n$$

$$u_{i,i-1} = \begin{cases} 0 & i = k \\ \dfrac{r_{ii} a_{i,i-1}}{r_{i-1,i-1}} & k+1 \le i \le n \end{cases}$$

$$v_{i-1,i} = \begin{cases} \dfrac{u_{k-1,k} R_k \bar{C}_k}{R_{k-1} C_{k-1}} & i = k \\ u_{i,i-1} & k+1 \le i \le n \end{cases}$$

$$u_{ii} = v_{ii} = \begin{cases} a_{kk} + \dfrac{r_{k,k+1} a_{k+1,k}}{r_{kk}} & i = k \\ \dfrac{c_{i-1,i} a_{i,i-1}}{c_{ii}} + a_{ii} + \dfrac{r_{i,i+1} a_{i+1,i}}{r_{ii}} & k+1 \le i \le n-1 \\ \dfrac{c_{n-1,n} a_{n,n-1}}{c_{nn}} + a_{nn} & i = n \end{cases}$$

<u>Corollary (1):</u>

The sequence of row vectors $R_1, \ldots, R_n$ and the tridiagonal matrices U and V can be constructed without constructing the sequence of column vectors $C_1, \ldots, \bar{C}_k, \ldots, C_n$.

Proof:

This follows from the formulae of this theorem and theorem (4.12). It is necessary to choose the arbitrary non-zero element $\bar{c}_{kk}$ and

$$c_{ii} = \prod_{j=k+1}^{i} \frac{a_{j,j-1} \bar{c}_{kk}}{v_{j,j-1}}.$$

But it is not even necessary to solve the system (4.16) since $\bar{c}_{kk}$ is the only element of $\bar{C}_k$ required to continue the algorithm.

Corollary (2):

The diagonal matrix $D = RC$ and the almost symmetric tridiagonal matrix DV can be constructed without constructing the sequence of column vectors $\{C_i\}$.

Modification in Case (B3)

As described in chapter 2 vectors $\overline{R}_k$ and $\overline{C}_k$ with the property (2.10) are chosen. $\overline{R}_k$ and $\overline{C}_k$ are chosen as in the modifications in cases (B1) and (B2) respectively and the algorithm is continued.

Theorem (4.18):

If for each pair of vectors R_i and C_i, $k+1 \le i \le n$, $R_i C_i \ne 0$ then the matrices

$$R = (R_1^T \cdots \overline{R}_k^T \cdots R_n^T)^T \quad \text{and} \quad C = (C_1 \cdots \overline{C}_k \cdots C_n)$$

are both upper triangular.

Proof:

Theorem (2.8) with corollaries (1''), (2''), and (3'') holds in this case. The proof proceeds as in theorem (4.1) and the following formulae are obtained:

$$\left.\begin{aligned} R_i &= (0, \cdots, 0, r_{ii}, \cdots, r_{in}) \\ C_i &= (c_{1i}, \cdots, c_{ii}, 0, \cdots, 0)^T \end{aligned}\right\} \quad k+1 \le i \le n$$

$$v_{k+1,k}c_{k+1,k+1} = a_{k+1,k}\overline{c}_{kk}$$

$$v_{i,i-1}c_{ii} = a_{i,i-1}c_{i-1,i-1} \qquad k+2 \le i \le n$$

$$\frac{\overline{c}_{k-1,k}}{\overline{c}_{kk}} = \frac{-\overline{r}_{k-1,k}}{r_{k-1,k-1}}$$

$$\frac{c_{k,k+1}}{c_{k+1,k+1}} = -\frac{\bar{r}_{k,k+1}}{\bar{r}_{kk}}$$

$$\frac{c_{i-1,i}}{c_{ii}} = -\frac{r_{i-1,i}}{r_{i-1,i-1}} \qquad k+2 \le i \le n$$

$$\bar{R}_k A \bar{C}_k = \bar{r}_{kk} a_{kk} \bar{c}_{kk} + \bar{r}_{k,k+1} a_{k+1,k} \bar{c}_{kk}$$

$$R_i A C_i = \begin{cases} r_{ii}(a_{i,i-1}c_{i-1,i} + a_{ii}c_{ii}) + r_{i,i+1}a_{i+1,i}c_{ii} & k+1 \le i \le n-1 \\ r_{nn}(a_{n,n-1}c_{n-1,n} + a_{nn}c_{nn}) & i = n \end{cases}$$

$$\bar{R}_k \bar{C}_k = \bar{r}_{kk}\bar{c}_{kk}$$

$$R_i C_i = r_{ii}c_{ii} \qquad k+1 \le i \le n$$

$$u_{k,k-1} = v_{k-1,k} = 0$$

$$u_{i,i-1} = \begin{cases} \dfrac{r_{k+1,k+1}}{\bar{r}_{kk}} a_{k+1,k} & i = k+1 \\[2ex] \dfrac{r_{ii}}{r_{i-1,i-1}} a_{i,i-1} & k+2 \le i \le n \end{cases}$$

$$v_{i-1,i} = u_{i,i-1} \qquad k+1 \le i \le n$$

$$u_{ii} = v_{ii} = \begin{cases} a_{kk} + \dfrac{\bar{r}_{k,k+1}}{\bar{r}_{kk}} a_{k+1,k} & i = k \\[2ex] \dfrac{c_{i-1,i}}{c_{ii}} a_{i,i-1} + a_{ii} + \dfrac{r_{i,i+1}}{r_{ii}} a_{i+1,i} & k+1 \le i \le n-1 \\[2ex] \dfrac{c_{n-1,n}}{c_{nn}} a_{n,n-1} + a_{nn} & i = n \end{cases}$$

Corollary (1):

The sequence of row vectors $R_1, \ldots, \bar{R}_k, \ldots, R_n$

and the tridiagonal matrices U and V can be constructed without constructing the sequence of column vectors $c_1, \ldots, \bar{c}_k, \ldots, c_n$.

Proof:

This follows from the formulae of this theorem and theorem (4.12). It is necessary to choose the arbitrary non-zero element $\bar{c}_{kk}$ but it is not necessary to solve the system (4.16). Also

$$c_{ii} = \prod_{j=k+1}^{i} \frac{a_{j,j-1}\bar{c}_{kk}}{v_{j,j-1}} .$$

Corollary (2):

The diagonal matrix $D = RC$ and the symmetric tridiagonal matrix DV can be constructed without constructing the sequence of column vectors.

Multiple Breakdowns

As was shown in chapter 2 unilateral and bilateral breakdowns may occur in any order and with any frequency and the algorithm can be completed by making the appropriate modification at each breakdown. Moreover, in the case where A is upper Hessenberg and the starting vectors are chosen as in theorem (4.12) it is unnecessary to construct the sequence of column vectors. The various coefficients are given by theorem (4.12) from the beginning of the algorithm until the first breakdown. After a breakdown of type (B1), (B2) or (B3) they are given by theorem (4.15), (4.17) or (4.18) respectively, but only as far as the next breakdown.

If no breakdowns or bilateral breakdowns only occur then a diagonal matrix and a symmetric tridiagonal matrix are constructed and if unilateral breakdowns occur then a diagonal matrix and an almost symmetric tridiagonal matrix are constructed.

Modification in Case (B4)

See this section in chapter 2.

Derogatory Matrix

In chapter 2 it was shown that breakdowns must occur if A is derogatory.

Theorem (4.19):

If $a_{i,i-1} \neq 0$ for $2 \leq i \leq n$ then breakdowns of type (B2) and (B3) cannot occur when $c_1 = (c_{11}, 0, \ldots, 0)^T$.

Proof:

It has been shown that $c_{ii} = \prod_{j=2}^{i} \frac{a_{j,j-1} c_{11}}{v_{j,j-1}}$

and so c_{ii} vanishes only if some $a_{i,i-1} = 0$.

Corollary:

A is non-derogatory.

Proof:

This follows from the corollary to theorem (2.11).

CHAPTER 5

SOLUTION OF THE MATRIX EIGENVALUE PROBLEM

In this chapter the results of chapters 1 to 4 are used to describe a method for finding the eigenvalues of a $n \times n$ matrix M. In general terms the method is as follows: transform M to an upper Hessenberg matrix H by orthogonal transformations, apply the method of minimized iterations to H to obtain a pair of symmetric matrices A and B, and apply the extended Rayleigh - Kohn method to the system $|A - rB| = 0$ to obtain its eigenvalues, which are just the eigenvalues of M.

Reduction to Hessenberg Form

An $n \times n$ matrix M can be reduced to upper Hessenberg form by means of $n-2$ orthogonal transformations described by Householder and Bauer [12] . These transformations are described in more detail by Wilkinson [13].

Let $T_2, \ldots , T_{n-1}$ denote the $n-2$ transformation matrices. The matrix T_{k+1} , $1 \leq k \leq n-2$, is of the form

(5.1)..... $T_{k+1} = I - 2w_{k+1}w_{k+1}^T$

where w_{k+1} is a column vector of the form

(5.2)..... $w_{k+1}^T = (0, \ldots , 0, w_{k+1,k+1} , \ldots , w_{k+1,n})$

(5.3)..... $w_{k+1}^T w_{k+1} = 1.$

It follows that $T_{k+1}^2 = I$ and therefore

(5.4)..... $T_{k+1} = T_{k+1}^T = T_{k+1}^{-1}$.

T_{k+1} can also be written in the form

(5.5)..... $T_{k+1} = \left[\begin{array}{c|c} I & 0 \\ \hline 0 & Q_{n-k} \end{array}\right]$ where

$$Q_{n-k} = I - 2w_{k+1}w_{k+1}^T \quad \text{and}$$

$$w_{k+1}^T = (w_{k+1,k+1}, \ldots, w_{k+1,n}).$$

Like T_{k+1}, Q_{n-k} has the properties

(5.6)..... $Q_{n-k} = Q_{n-k}^T = Q_{n-k}^{-1}$.

Let $M_1 = M$ and let $M_2, \ldots, M_{n-1} = H$ denote the transformed matrices. The reduction to Hessenberg form is described by

$$M_{k+1} = T_{k+1}M_kT_{k+1} \qquad 1 \le k \le n-2.$$

Let

(5.7)..... $M_k = \left[\begin{array}{c|c} A_k & C \\ \hline Z & B_{n-k} \end{array}\right]$

where A_k is a $k \times k$ submatrix. Then, using (5.5),

(5.8)..... $M_{k+1} = T_{k+1}M_kT_{k+1} = \left[\begin{array}{c|c} A_k & CQ_{n-k} \\ \hline Q_{n-k}Z & Q_{n-k}B_{n-k}Q_{n-k} \end{array}\right]$.

If the first k-1 columns of M_k are in upper Hessenberg form then the first k-1 columns of Z, and therefore of $Q_{n-k}Z$, are zero. Therefore the transformation by T_{k+1} preserves the partial Hessenberg form of M_k. w_{k+1} is chosen so that the Hessenberg form is extended to column k of M_{k+1}, that is so

(5.9)..... $Q_{n-k}Z' = Q_{n-k}((M_k)_{k+1,k}, \ldots, (M_k)_{nk})^T$

$$= ((M_{k+1})_{k+1,k}, 0, \ldots, 0)^T$$

where Z' is the last column of Z. From (5.5)

$$(5.10)\ldots\ldots \quad Q_{n-k}Z' = Z' - 2V_{k+1}V_{k+1}^{T}Z'$$
$$= Z' - 2V_{k+1}q \qquad \text{where}$$

$$q = \sum_{i=k+1}^{n} w_{k+1,i}(M_k)_{ik} \ .$$

Therefore

$$(5.11)\ldots\ldots \quad Q_{n-k}Z' = ((M_k)_{k+1,k} - 2qw_{k+1,k+1}, \ldots, (M_k)_{nk} - 2qw_{k+1,n})^{T}.$$

Imposing the condition of (5.9) on the right hand side of (5.11) yields

$$(5.12)\ldots\ldots \quad (M_k)_{ik} - 2qw_{k+1,i} = 0 \qquad k+2 \le i \le n.$$

Let

$$(5.13)\ldots\ldots \quad S = \sum_{i=k+1}^{n} (M_k)_{ik}^2.$$

Using (5.7), (5.11), and (5.12) it follows that

$$S = Z'^{T}Z' = Z'^{T}Q_{n-k}Q_{n-k}Z'$$
$$= (Q_{n-k}Z')^{T}Q_{n-k}Z' = ((M_k)_{k+1,k} - 2qw_{k+1,k+1})^2.$$

Therefore

$$(5.14)\ldots\ldots \quad (M_k)_{k+1,k} - 2qw_{k+1,k+1} = \pm\sqrt{S}$$

The n-k equations of (5.12) and (5.14) are modified and used to determine the n-k elements of w_{k+1}. This is done by multiplying each equation in (5.12) by $w_{k+1,i}$ and (5.14) by $w_{k+1,k+1}$. The resulting n-k equations are summed and (5.3) is used to obtain

(5.15)..... $q = \mp w_{k+1,k+1}\sqrt{S}$.

(5.15) is substituted into (5.14) and (5.12) to obtain

$$(5.16)\ldots\ldots \begin{cases} w^2_{k+1,k+1} = \frac{1}{2}\left[1 \mp ((M_k)_{k+1,k}/\sqrt{S})\right] \\ w_{k+1,i} = \mp (M_k)_{ik} / (2w_{k+1,k+1}\sqrt{S}) \qquad k+2 \le i \le n \end{cases}$$

which determine the elements of W_{k+1}. The upper or lower sign must be used throughout the calculations and this sign should be chosen so that $w_{k+1,k+1}$ is as large as possible.

If the matrix M is symmetric then M_{k+1} is symmetric, $1 \le k \le n-2$, and the resulting Hessenberg matrix is a tridiagonal symmetric matrix. In this case the extended Rayleigh - Kohn method can be applied directly to $|H - rI| = 0$, provided H is non-derogatory.

If M is derogatory, symmetric or not, then H is derogatory. If there are no zeros in the lower codiagonal of H then according to the corollary of theorem (4.19) H is non-derogatory. Therefore if M is derogatory there is at least one zero in the lower codiagonal of H.

Thus in the case in which M is symmetric and derogatory the matrix H is a direct sum of non-derogatory tridiagonal symmetric matrices. The extended Rayleigh - Kohn method can be applied to each of the smaller non-derogatory matrices. In any case, if the original matrix M is symmetric the method of minimized iterations is not required in finding the eigenvalues of M.

Lanczos' Algorithm

In chapter 4 the application of Lanczos' algorithm to a Hessenberg matrix H is described. This algorithm produces a tridiagonal matrix V which is similar to H and a diagonal matrix D such that DV is symmetric or almost symmetric in the following sense:

$$(5.27)\ldots\ldots \begin{cases} \text{if } (DV)_{i+1,i} \neq (DV)_{i,i+1} \text{ then one} \\ \text{of the above pair is zero.} \end{cases}$$

If DV is symmetric then according to the corollary of theorem (4.19) DV is the direct sum of one or more non-derogatory matrices. Each of the non-derogatory sub-matrices in $|DV - rD| = 0$ can be solved for its eigenvalues by the extended Rayleigh - Kohn method.

If DV is almost symmetric, the eigenvalues of H are the roots of $|DV - rD| = 0$ but the extended Rayleigh - Kohn method cannot be applied to this since DV is not symmetric. Let $P = (p_{ij}) = DV - rD$ and let P_i denote the i'th principal minor of P. It is easily shown by induction that, since P is tridiagonal,

$$P_1 = p_{11}$$

$$P_{j+1} = p_{j+1,j+1}P_j - p_{j+1,j}p_{j,j+1}P_{j-1} \qquad 1 \leq j \leq n-1.$$

Therefore if one of $(DV)_{i+1,i}$ and $(DV)_{i,i+1}$ is zero then $p_{i+1,i}p_{i,i+1} = 0$, $(DV)_{i+1,i}$ and $(DV)_{i,i+1}$ do not appear in P_{i+1} , and therefore they do not appear in $P_n = |P|$. For each pair of elements in DV which have the property (5.27) one may replace the non-zero element of the pair by zero and denote the resulting matrix by DV^*. Then $|DV^* - rD| = |DV - rD|$ and DV^* is symmetric. Moreover, DV^* is the direct sum of non-derogatory tridiagonl symmetric matrices and the extended Rayleigh - Kohn iteration can be used.

Solution of $|A - rB| = 0$

The extended Rayleigh - Kohn iteration is described in chapter 1. However, some farther remarks should be made concerning it. When starting the solution of $|A - rB| = 0$ the vector R, appearing in the right hand side vector BR in (1.15), should have a component in the direction of each eigenvector associated with a simple root and a component in the direction of at least one of the principal vectors P_j, $t \leq j \leq n$ where t is defined by (1.19), for each invariant subspace associated with a multiple root. In practice one can only choose R arbitrarily, say $R = (1, \ldots, 1)^T$. With the finding of each root the vector R is modified to prevent reconvergence to this root. Let R' denote the modified R. It may happen that at some stage R' = 0 before all the roots are found. This indicates that not all the invariant subspaces were represented in the original R. In this case a new R should be chosen which, when modified to prevent convergence to known roots, is non-zero. It may also happen that for one or more invariant subspaces, associated with a multiple root, only the eigenvectors or principal vectors P_j such that $j<t$ are represented in R. This state of affairs will manifest

itself when $R'^TBR' = 0$ after all the other eigenvalues have been found and R has been modified accordingly. This is a result of the orthogonality conditions described in (1.10) and (1.13). In this case a new R must be chosen which, when modified to prevent convergence to known roots, does not have the property $R'^TBR' = 0$.

The equation $|A - rB| = 0$ may well have pairs of complex roots. To find these, complex initial estimates must be made or a complex right hand side chosen. This can be done when, with a real right hand side, the method does not converge.

As w_j converges to an eigenvalue the matrix $(A - w_jB)$ may become exactly singular, in which case an eigenvalue has been found. This is discovered when one tries to solve $(A - w_jB)X_{j+1} = BR$ for X_{j+1}. In this case the vector X_j from which w_j was formed can be taken as the eigenvector or $(A - w_jB)X_{j+1} = 0$ can be solved for the eigenvector. A good elimination procedure for solving the tridiagonal system $(A - w_jB)X_{j+1} = BR$ when the matrix is nearly singular is described by Wilkinson [14].

As described in chapter 1, if $X_j^TBX_j$ becomes arbitrarily small as w_j converges then the root being found is multiple. As w_j converges $(A - w_jB)$ becomes increasingly singular and the elements of X_j become larger. Thus to determine, in practice, if the root is multiple the quadratic form $S_j^T B S_j$, where S_j is X_j suitably scaled, should be tested. S_j might be obtained by scaling the first element of X_j or the largest element of X_j to 1.

To determine the multiplicity of a root the additional systems

$$(A - w_{j-1}B)V_j^{(k)} = BV_j^{(k-1)}$$

are solved for $k = 1,2,3$, etc., $V_j^{(0)} = X_j$, until the first system for which $X_j^T B V_j^{(k)}$ does not become arbitrarily small is found. This bilinear form should be computed only after X_j and $V_j^{(k)}$ have been scaled. Note that the system $(A - w_{j-1}B)V_j^{(1)} = BV_j^{(0)}$ cannot be solved by elimination as can the system $(A - w_{j-1}B)X_j = BR$ since those arguments which show that X_j is a good approximation to the eigenvector also show that $V_j^{(1)}$ will be a good approximation to the eigenvector. The systems $(A - w_{j-1}B)V_j^{(k)} = BV_j^{(k-1)}$ should be solved by setting the first k components of $V_j^{(k)}$ to zero and solving for the remaining components by recursion. The justification for this is the fact that the k'th principal vector associated with a multiple root is zero in the first k components. This can be seen by examining the following equation which is assumed to have a root r of multiplicity 3.

$$\begin{vmatrix} a_{11} - rb_1 & a_{12} & \\ a_{12} & a_{22} - rb_2 & a_{23} \\ & a_{23} & a_{33} - rb_3 \end{vmatrix} = |A - rB| = 0.$$

Let X, Y, Z, denote the eigenvector and the first and second principal vectors respectively. Let $M_i(r)$ denote the i'th principal minor of the above determinant. It follows that

$$M_1(r) = a_{11} - rb_1$$

$$M_2(r) = (a_{22} - rb_2)M_1(r) - a_{12}^2$$

$$M_3(r) = (a_{33} - rb_3)M_2(r) - a_{23}^2 M_1(r) .$$

Since r is a root of multiplicity 3

$$M_3(r) = M_3'(r) = M_3''(r) = 0$$

Using this fact it is easy to show that the solutions of the following systems

$$(A - rB)X = 0$$

$$(A - rB)Y = BX$$

$$(A - rB)Z = BY$$

are

$$X = \left[1, -\frac{M_1(r)}{a_{12}}, \frac{M_2(r)}{a_{12}a_{23}} \right]^T$$

$$Y = \left[0, -\frac{M_1'(r)}{a_{12}}, \frac{M_2'(r)}{a_{12}a_{23}} \right]^T$$

$$Z = \left[0, 0, \frac{M_2''(r)}{2a_{12}a_{23}} \right]^T$$

which when scaled are

$$X = (1, *, *)^T$$

$$Y = (0, 1, *)^T$$

$$Z = (0, 0, 1)^T.$$

CHAPTER 6

TEST RESULTS

In this chapter results obtained in two test cases of low order, for which the exact eigenvalues are known, are described. Also, some of the difficulties encountered with other test cases are briefly discussed.

Test Case 1

This matrix M was taken from [9] and the exact eigenvalues are 1,2, 5, 8, 9.

$$M = \begin{bmatrix} 24 & -23 & 6 & -21 & 27 \\ 3 & -2 & -26 & 59 & 49 \\ -1 & 1 & 11 & -24 & -20 \\ 2 & -2 & -17 & 38 & 31 \\ -3 & 3 & 26 & -56 & -46 \end{bmatrix}$$

Householder's transformations transformed M to the upper Hessenberg matrix

$$H = \begin{bmatrix} 24.0000 & 41.2859 & 2.6673 & 3.9196 & 2.8284 \\ -4.7958 & -3.0435 & -43.5022 & 108.2612 & -51.1624 \\ & .2039 & 2.4071 & -5.5894 & 2.8291 \\ & & .3083 & -.3636 & .6396 \\ & & & -2.3452 & 2.0000 \end{bmatrix}$$

to 4 decimal places. The method of minimized iterations with starting vectors $R_1 = (1, \dots , 1)$ and $C_1 = (1, 0, \dots , 0)^T$ and scale factors chosen so $R_1C_1 = \pm 1$ produced the following symmetric tridiagonal matrix

$$A = \begin{bmatrix} 19.2042 & -9.6064 & & & \\ -9.6064 & -1.1448 & 2.8955 & & \\ & 2.8955 & 2.0350 & -.7827 & \\ & & -.7827 & -2.2774 & .9991 \\ & & & .9991 & .3386 \end{bmatrix}$$

to 4 decimal places, and the following diagonal matrix

$$B = \begin{bmatrix} 1 & & & & \\ & -1 & & & \\ & & 1 & & \\ & & & -1 & \\ & & & & 1 \end{bmatrix}$$

The extended Rayleigh - Kohn method with starting right hand vector $R = (1, \ldots, 1)^T$ was applied to $(A - rB)$ and the following results were obtained:

No.	Initial Estimate of Eigenvalue	No. of Iterations	Computed Eigenvalue
1	3.	8	1.0000
2	3.	6	2.0000
3	3.	6	5.0000
4	3.	8	8.0000
5	3.	1	9.0000

The computed eigenvalues were all correct to at least 10 digits when a word length of 12 digits was used.

<u>Test Case 2</u>

This matrix

$$M = \begin{bmatrix} 12 & 20 & -12 & 8 & -7 \\ -2 & 9 & -2 & -1 & -3 \\ 14 & 26 & -14 & 10 & -9 \\ -6 & -14 & 8 & -4 & 5 \\ -22 & -8 & 12 & -14 & 3 \end{bmatrix}$$

has the exact eigenvalues -1, 0, 2, 2, 3. Moreover M is derogatory. In this case

$$H = \begin{bmatrix} 12.0000 & -3.8013 & 2.4189 & -9.1570 & -23.5127 \\ 26.8328 & -8.8333 & 6.3314 & -18.4911 & -27.9225 \\ & -.6625 & 1.8333 & 2.5566 & 21.7226 \\ & & 0.0000 & -.9379 & -11.6385 \\ & & & -.0157 & 1.9379 \end{bmatrix}$$

As was expected in this derogatory case, a zero appeared in the lower codiagonal of H. In actual floating point computation the (4,3) element of H was found to be of the order of 10^{-13} when a word length of 16 digits was used. Before applying the method of minimized iterations this element was set to zero. The following almost symmetric tridiagonal and diagonal matrices were obtained.

$$A = \begin{bmatrix} 38.8328 & -37.4004 & & & \\ -37.4004 & 36.0252 & .0964 & & \\ & .0964 & 2.1923 & -14.1244 & \\ & & 0 & -1.0000 & 0 \\ & & & -.0157 & 2.0000 \end{bmatrix}$$

$$B = \begin{bmatrix} 1 & & & & \\ & -1 & & & \\ & & 1 & & \\ & & & 1 & \\ & & & & 1 \end{bmatrix}$$

Due to the zero in the lower codiagonal of H there was a unilateral breakdown at that point in the C vectors and this produced a zero in the same position in A. Since M and hence H are derogatory a breakdown in the B vectors was expected also. The vector R_5 did not vanish but its elements were of the order of 10^{-12} when the word length was 16 digits so it was considered to be zero. This produced the zero in the upper ~~endiagonal~~ of A. The eigenvalues of the system $(A - rB)$ are unchanged if the (3,4) and (5,4) elements of A are set to zero. When this is done the problem of finding the eigenvalues is split into that of finding the eigenvalues of a 3×3 and two 1×1 systems. Therefore the eigenvalues -1 and 2 were obtained at this point. They were found accurately to 12 digits. The extended Rayleigh - Kohn method when applied to the remaining 3×3 system gave the following results:

No.	Initial Estimate of Eigenvalue	No. of Iterations	Computed Eigenvalue
1	5.	6	3.0000
2	5.	4	0.0000
3	5.	1	2.0000

The second root found was not identically zero but of the order of 10^{-9} when the word length was 12 digits. The other two roots were as accurate as those in test case 1.

Conclusion

The first test case seems to indicate that if the eigenvalues are simple and well separated the method works well. However if the original matrix H is derogatory then, in practice, a very small element appears in the lower codiagonal of H and some R vector in the method of minimized iterations is very small. This creates the problem of deciding if an element or vector is small enough to be considered zero. This problem might be handled by choosing a number e, based on experience with test cases, and considering elements whose norms are less than e to be zero.

Another problem which arises and which is not encountered in the previous two test cases is the following. When $(A - rB)$ has a multiple root and w_j is converging to this root then $X_j^T B X_j$ becomes very small and since it is a sum of numbers of ordinary size the smaller it becomes the fewer significant digits it contains until a point is reached beyond which it is impossible to improve the estimate w_j. This problem might be overcome by computing $X_j^T B X_j$ and possibly even X_j to higher precision. Another problem which occurs simultaneously with the preceding problem is that of determining the multiplicity of a root. Theoretically one solves the additional systems described in chapter 1 until the first is found for which $X_j^T B V_j^{(k)}$ does not become arbitrarily small. Here again some rule must be chosen for deciding when $X_j^T B V_j^{(k)}$ is not small.

It appears that the method will work well in those cases in which the eigenvalues are simple and well spaced. In those cases in which the eigenvalues are multiple the Householder transformations and the method of minimized iterations should be carried out very accurately to prevent multiple eigenvalues from appearing as nearly equal eigenvalues. If in addition higher precision is used at least in some parts of the extended Rayleigh - Kohn method this method may also work reasonably well when the original matrix has multiple eigenvalues.

REFERENCES

[1] Ostrowski A.M. On the Convergence of the Rayleigh Quotient Iteration for the Computation of the Characteristic Roots and Vectors III and IV. Archive for Rational Mechanics and Analysis (1959) 325.

[2] Dessautels E. Symmetrizing Matrices. Thesis, University of Ottawa (1963).

[3] Crandall S.H. Iterative Procedures Related to Relaxation Methods for Eigenvalue Problems. Proc. Royal Soc. A 207 (1951) 418 - 420.

[4] Ostrowski A.M. On the Convergence of the Rayleigh Quotient Iteration for the Computation of the Characteristic Roots and Vectors I. Archive for Rational Mechanics and Analysis (1958) 233.

[5] Howland J.L. On Some Methods for Computing the Roots of Polynomials. Proc. IFIP Congress Munich (1962).

[6] Lanczos C. An Iteration Method for the Solution of the Eigenvalue Problem of Linear Differential and Integral Operators. Jour. Res. N.B.S. 45 No. 4 (1950) 255-282.

[7] Causey R.L. and Gregory R.T. On Lanczos' Algorithm for Tridiagonalizing Matrices. SIAM Review 3 (1961) 322 - 328.

[8] Rutishauser H. Beitrage zur Kenntnis des Biorthogonalisierungs Algorithmus von Lanczos. Z. Angew. Math Physik 4 (1953) 35 -56.

[9] Durand E. Solutions Numeriques des Equations Algebriques. Tome 2, Masson et Cie. (1962).

[10] Faddeev D.K. and Faddeeva V.N. Computational Methods of Linear Algebra. Freeman (1963) 368 - 369.

[11] Wilkinson J.H. Symposium on Methods for Solving Linear Systems. Proc. Int. Conf. on Inf. Proc. UNESCO Paris (1959) 108.

[12] Householder A.S. and Bauer F.L. On Certain Methods for Expanding the Characteristic Polynomial. Num. Math. 1 (1959) 29 - 37.

[13] Wilkinson J.H. Householder's Method for the Solution of Algebraic Eigenproblem. Comp. Jour. 3 1 (1960) 23 - 27.

[14] Wilkinson J.H. The Calculation of the Eigenvectors of Codiagonal Matrices. Comp. Jour. 1 2 (1958) 92.