3. Solution of Eigenvalue Problems With the LR-Transformation[1,2]

Heinz Rutishauser[3]

Contents

Page
1. Introduction 47
2. Properties of the LR-transformation 49
3. Convergence of A_k for $k\to\infty$ 50
4. Numerical examples 53
    4.1. Wilson's matrix (turned upside down) 53
    4.2. Matrix with double latent roots 53
    4.3. Matrix with "disorder of latent roots" 54
5. The case of a real matrix with complex latent roots 55
6. Improvement of convergence 58
    6.1. The case of real latent roots 58
    6.2. The case of conjugate complex latent roots 62
    6.3. Special remarks 64
7. Determination of latent vectors 66
8. Corrective measures against roundoff errors, estimates 67
9. Connections with "deflation" 68
10. Determination of latent roots of striped matrices 71
11. A continuous analog to the LR-transformation 73
12. A Graeffe-like modification of the LR-transformation 75
13. Appendix. Numerical experiments with striped matrices 78
14. References 80

1. Introduction

In an earlier paper [7] [4] it was shown that the progressive form of the quotient-difference algorithm [6, section 5] is a special case of a more general method—called LR-transformation—which permits the determination of latent roots and vectors of matrices. Whereas [7] was only a preliminary report on the subject, the present paper gives full details and proofs, as well as an extended form of the LR-transformation. The method can be described as follows:

Starting with the given matrix $A=A_1$, we compute the triangular decomposition [5] $A_1=L_1R_1$, where

$$L_1=\begin{bmatrix} 1 & & & & & \\ * & 1 & & \mathbf{O} & & \\ * & * & 1 & & & \\ * & * & * & \cdot & & \\ * & * & * & & \cdot & \\ * & * & * & * & \cdot & \\ * & * & * & * & * & 1 \end{bmatrix}, \qquad R_1=\begin{bmatrix} r_{11} & * & * & * & * & * \\ & r_{22} & * & * & * & * \\ & & \cdot & * & * & * \\ & \mathbf{O} & & \cdot & * & * \\ & & & & \cdot & * \\ & & & & & r_{nn} \end{bmatrix}, \tag{1}$$

---

[1] (a) In this report the expressions "latent root" and "latent vector" have been used throughout in place of eigenvalue and eigenvector; (b) the unit matrix is denoted by E; and (c) the matrix $(\lambda_i\delta_i)$, where $\delta_{ij}=0$ if $i\neq j$, $\delta_{ij}=1$ if $i=j$, is denoted by diag $(\lambda_1, \lambda_1, \ldots, \lambda_n)$.

[2] Section 12 and parts of section 2 have been worked out in cooperation with F. L. Bauer of the Technische Hochschule, München, Germany (Department of Mathematics).

[3] Eidgenossische Technische Hochschule, Zürich.

[4] Figures in brackets indicate the literature references at the end of this paper.

[5] Methods to obtain such a decomposition are well known, for instance the mechanized Gaussian algorithm as described by Zurmühl [10, section 1.3]. The ordinary Gauss-Banachiewicz procedure, however, prescribes 1's as diagonal elements of R and is therefore transposed to the elimination scheme used here.

and then multiply L_1 and R_1 in the reversed order: $A_2=R_1L_1$. This gives a new matrix A_2 having the same latent roots as A_1, because obviously $A_2=R_1A_1R_1^{-1}$. If A_2 is treated in the same way as A_1, a sequence of matrices $A_1A_2A_3$ . . . is obtained. Under certain conditions A_k converges for $k\rightarrow\infty$ to an upper triangular matrix which has the latent roots of A as diagonal elements.

The QD algorithm corresponds to the case where A is a Jacobi matrix:

$$A_1=J_0=\begin{pmatrix} a_1 & 1 & & & & \\ b_1 & a_2 & 1 & & \mathbf{O} & \\ & b_2 & a_3 & 1 & & \\ & & \cdot & \cdot & \cdot & \\ & & & \cdot & \cdot & \cdot \\ & \mathbf{O} & & & \cdot & 1 \\ & & & & b_{n-1} & a_n \end{pmatrix}$$

Then the decomposition of J_k in L_k and R_k obviously corresponds to the formulas (8) in [6, section 7] (with upper index $k=\nu+1$), whereas the formulas (7) (with $k=\nu$) describe the operation $R_k\cdot L_k=J_{k+1}$.

The method seems very time-consuming on first sight. However, triangular decomposition is a very simple process and allows easy checks. Moreover, problems in numerical analysis often lead to the determination of latent roots of striped matrices;

$$A=(a_{ij}) \quad \text{with} \quad a_{ij}=0 \quad \text{for} \quad |i-j|>m, \tag{2}$$

which, for $m=1$, include the Jacobi matrices. Clearly, property (2) is maintained by the LR-transformation; i.e., if A is of that form, all matrices A_k will have the same property with the same value of m. This results in a great saving in computing time. For instance, the number of multiplications and divisions together needed for triangularization of an n-row matrix is reduced from $(n^3-n)/3$ to $[m(m+1)(3n-2m-1)]/3$, so that the LR-transformation is especially well suited for such matrices. Note that property (2) is destroyed by the method of Jacobian rotations (see Jacobi [3] or Gregory [2]).

Numerical experiments have shown that the LR method does not fail for such matrices as A and B below:

$$A=\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 3 & 6 & 10 & 15 & 21 \\ 1 & 4 & 10 & 20 & 35 & 56 \\ 1 & 5 & 15 & 35 & 70 & 126 \\ 1 & 6 & 21 & 56 & 126 & 252 \end{pmatrix} \quad \begin{array}{l} \lambda_1=1/\lambda_6=332.84\ .\ .\ . \\ \lambda_2=1/\lambda_5=\ 15.553\ .\ .\ . \\ \lambda_3=1/\lambda_4=\ 2.0435\ .\ .\ . \end{array}$$

$$B=\begin{pmatrix} 7 & -14 & 21 & -14 & 7 & 0 \\ -14 & 57 & -82 & 73 & -24 & 11 \\ 21 & -82 & 152 & -117 & 71 & 11 \\ -14 & 73 & -117 & 137 & -19 & 66 \\ 7 & -24 & 71 & -19 & 96 & 121 \\ 0 & 11 & 11 & 66 & 121 & 253 \end{pmatrix} \quad \begin{array}{l} \lambda_1=\lambda_2=332.84\ .\ .\ . \\ \lambda_3=\lambda_4=\ 15.617\ .\ .\ . \\ \lambda_5=\lambda_6=\ 2.5329\ .\ .\ . \end{array}$$

A has reciprocal pairs of latent roots, which is a general property of matrices with

$$a_{ij}=\binom{i+j-2}{i-1} \qquad (i,j=1,2,\ .\ .\ .\ ,n)$$

because there exists a matrix C with $A=CC^T$; $A^{-1}=C^TC$. From the properties of A we infer that $B=A+A^{-1}$ has three pairs of equal latent roots. The characteristic polynomial of B is

$$(\lambda^3-351\lambda^2+6081\lambda-13167)^2.$$

2. Properties of the LR-Transformation

Let $A=A_1$ be a square matrix of order n and L_1R_1 its decomposition in a lower and upper triangular matrix, with the additional condition that the diagonal elements of L_1 be 1's.[6] It may be noted that the nontrivial elements l_{ik} and r_{ij} of the matrices L_1 and R_1 are computed by the following recursion formulas:

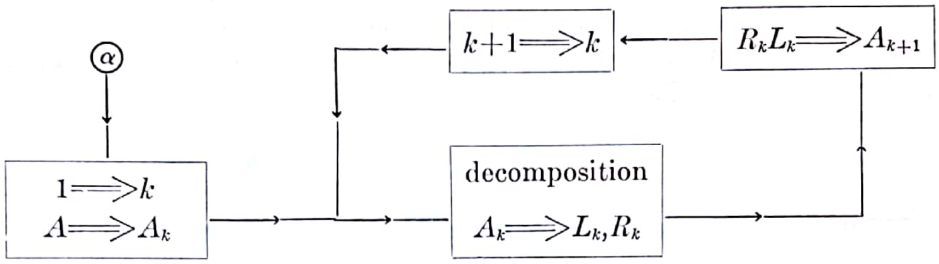
$$\left.\begin{array}{ll} r_{ij}=a_{ij}-\sum_{1}^{i-1} l_{ip}r_{pj} & \text{for}\quad i=1,2,\ldots,j \\ l_{ij}=\dfrac{a_{ij}-\sum_{1}^{j-1} l_{ip}r_{pj}}{r_{jj}} & \text{for}\quad i=j+1,\ldots,n \end{array}\right\}\quad \text{for}\quad j=1,2,\ldots,n. \tag{3}$$

p. 80.
Comp. Methods of
Lin. Algebra
V. N. Faddeeva

Then we multiply L_1 and R_1 together in the reversed order:[7]

$$R_1L_1 \Longrightarrow A_2. \tag{4}$$

By repeating the procedure (3), (4), we obtain an iterative process which yields an infinity of matrices A_k:



(5)

We have already stated *that A_k converges for $k\to\infty$, and under certain conditions A_k converges to an upper triangular matrix A_∞, whose diagonal elements are the latent roots of A.* This will be proved in section 3. Here we discuss only some properties of the matrices $A_kL_kR_k$. Since

$$A_2=R_1L_1=L_1^{-1}L_1R_1L_1=L_1^{-1}A_1L_1,$$

$$A_3=R_2L_2=L_2^{-1}A_2L_2=(L_1L_2)^{-1}A_1L_1L_2,$$

and so on, we see that

(a) All the A_k have the same latent roots; more exactly, they are all similar.

(b) The products

$$\Lambda_k=L_1L_2\ldots L_k \quad\text{and}\quad \mathrm{P}_k=R_kR_{k-1}\ldots R_2R_1 \tag{6}$$

are transformation matrices which transform A_1 into A_{k+1}:

$$A_{k+1}=\Lambda_k^{-1}A_1\Lambda_k=\mathrm{P}_kA_1\mathrm{P}_k^{-1}. \tag{7}$$

Λ_k and P_k are still left and right triangular matrices, with Λ_k having ones as diagonal elements. We now form the product

$$\Lambda_k\mathrm{P}_k=L_1L_2\ldots L_{k-1}L_kR_kR_{k-1}\ldots R_2R_1. \tag{8a}$$

With the general rule $L_xR_x=R_{x-1}L_{x-1}$, (8a) is converted into

$$\Lambda_k\mathrm{P}_k=L_1L_2\ldots L_{k-2}L_{k-1}R_{k-1}L_{k-1}R_{k-1}R_{k-2}\ldots R_2R_1, \tag{8b}$$

and by further applications of that rule into $(L_1R_1)^k=A^k$. But since Λ_k was a left and P_k a right triangular matrix, we have

---

[6] Like formula (1). In the sequel, we shall always use this type of decomposition.

[7] For the meaning of the symbol $\Longrightarrow$, see [9].

THEOREM 1. *The matrices Λ_k and P_k as defined in* (6) *can be obtained by triangular decomposition*[8] *of A^k:*

$$A^k = \Lambda_k \cdot \mathrm{P}_k. \tag{9}$$

3. Convergence of A_k for $k \to \infty$

As a first step we prove

THEOREM 2. *If the matrices Λ_k as defined in* (6) *converge for $k \to \infty$, then $\lim\limits_{k\to\infty} A_k$ exists and is an upper (right) triangular matrix A_∞.*

PROOF. If $\Lambda_\infty = \lim\limits_{k\to\infty} \Lambda_k$ exists, then $\lim L_k = \lim \Lambda_{k-1}^{-1}\Lambda_k = E$ (unit matrix) and

$$R_\infty = \lim R_k = \lim \Lambda_k^{-1} A_1 \Lambda_{k-1} = \Lambda_\infty^{-1} A \Lambda_\infty$$

exists also; therefore,

$$A_\infty = \lim A_k = \lim L_k R_k = R_\infty$$

exists too and is triangular.

We now investigate the conditions under which Λ_k converges for $k \to \infty$.

If A has a decomposition $A = LR$, then there exist explicit formulas for L and R. Let a_{ij} be the elements of A and l_{ij} those of L. Then

$$l_{ij} = \frac{D_{ji}}{D_{jj}}, \qquad D_{ji} = \begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1j} \\ a_{21} & a_{22} & \cdots & a_{2j} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ a_{j-1,1} & a_{j-1,2} & & a_{j-1,j} \\ a_{i1} & a_{i2} & \cdots & a_{ij} \end{vmatrix}. \tag{10}$$

If we want the decomposition of A^k, we have to replace the a_{ij} in (10) by the elements $a_{ij}^{(k)}$ of A^k; then (10) gives us the elements of Λ_k. Provided the elementary divisors of A are all linear, i. e., when A can be transformed into diagonal form:

P79 Bodewig. "Matrix Calculus"

$$A = U \operatorname{diag}(\lambda_1, \ldots, \lambda_n)\, U^{-1},$$

then we have

$$a_{ij}^{(k)} = \sum_{s=1}^{n} u_{is} v_{js} \lambda_s^k, \tag{11}$$

where u_{is} are the elements of U, and v_{js} those of U^{-1T}.

With (11), the matrix of D_{ji} in (10) can be written as the product of two rectangular matrices,

$$\begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ u_{21} & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & & \vdots \\ u_{j-1,1} & u_{j-1,2} & \cdots & u_{j-1,n} \\ u_{i1} & u_{i2} & \cdots & u_{in} \end{pmatrix} \text{ and } \begin{pmatrix} v_{11}\lambda_1^k & \cdots & v_{j1}\lambda_1^k \\ v_{12}\lambda_2^k & \cdots & v_{j2}\lambda_2^k \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ v_{1n}\lambda_n^k & \cdots & v_{in}\lambda_n^k \end{pmatrix}, \tag{12}$$

---

[8] See footnote 6.

and therefore D_{ji}—after a well-known theorem—is the sum of the products of corresponding j-rowed minors of the two matrices (12), the sum being extended over all possible combinations of j of the n numbers $1, 2, \ldots, n$.

This makes

$$l_{ij}=\frac{\sum u^{(i)}_{\alpha_1\alpha_2\ldots\alpha_j} v_{\alpha_1\alpha_2\ldots\alpha_j}(\lambda_{\alpha_1}\lambda_{\alpha_2}\cdots\lambda_{\alpha_j})^k}{\sum u_{\alpha_1\alpha_2\ldots\alpha_j} v_{\alpha_1\alpha_2\ldots\alpha_j}(\lambda_{\alpha_1}\lambda_{\alpha_2}\cdots\lambda_{\alpha_j})^k}, \tag{13}$$

where l_{ij} are now the elements of Λ_k and $u^{(i)}_{\alpha_1\ldots\alpha_j}$ is the j-rowed minor formed by the rows $1, 2, \ldots, j-1, i$ and columns $\alpha_1\alpha_2\ldots\alpha_j$ of U; moreover, $u_{\alpha_1\alpha_2\ldots\alpha_j}$ is the minor formed by the rows $1, 2, \ldots, j-1, j$ and columns $\alpha_1\alpha_2\ldots\alpha_j$ of U. $v_{\alpha_1\alpha_2\ldots\alpha_j}$ is the corresponding minor of the matrix V.

From this we see at once:

THEOREM 3. *If the latent roots λ_i of A and the matrices U and V defined in* (11) *fulfill the conditions*

$$\left.\begin{array}{l}
\text{(a)}\quad |\lambda_1|>|\lambda_2|>|\lambda_3|>\cdots>|\lambda_n|, \\[2ex]
\text{(b)}\quad \begin{vmatrix} u_{11} & \cdots & u_{1j} \\ \vdots & & \vdots \\ u_{j1} & \cdots & u_{jj} \end{vmatrix}\begin{vmatrix} v_{11} & \cdots & v_{1j} \\ \vdots & & \vdots \\ v_{j1} & \cdots & v_{jj} \end{vmatrix} \neq 0 \quad \text{for} \quad j=1,2,\ldots,n,
\end{array}\right\} \tag{14}$$

then $\lim_{k\to\infty} \Lambda_k=\Lambda_\infty$ *exists and is the lower triangle of the triangular decomposition of the matrix U, and*

$$\lim_{k\to\infty} A_k=A_\infty=\begin{pmatrix} \lambda_1 & * & * & * & * & * \\ & \lambda_2 & * & * & * & * \\ & & \cdot & * & * & * \\ & & & \cdot & * & * \\ & \mathbf{0} & & & \cdot & * \\ & & & & & \lambda_n \end{pmatrix}. \tag{15}$$

PROOF. Under conditions (14) the dominant terms in the denominator and numerator of (13) are

$$\begin{vmatrix} u_{11} & \cdots & u_{1j} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ u_{j1} & \cdots & u_{jj} \end{vmatrix}\begin{vmatrix} v_{11} & \cdots & v_{1j} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ v_{j1} & \cdots & v_{jj} \end{vmatrix}(\lambda_1\lambda_2\cdots\lambda_j)^k \quad \text{and} \quad \begin{vmatrix} u_{11} & \cdots & u_{1j} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ u_{j-1,1} & \cdots & u_{j-1,j} \\ u_{i1} & \cdots & u_{ij} \end{vmatrix}\begin{vmatrix} v_{11} & \cdots & v_{1j} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ v_{j1} & \cdots & v_{jj} \end{vmatrix}(\lambda_1\lambda_2\cdots\lambda_j)^k,$$

respectively, so that we have the following expressions for the elements of the matrix Λ_∞:

$$\lim_{k\to\infty} l^{(k)}_{ij}=\begin{vmatrix} u_{11} & u_{12} & \cdots & u_{1j} \\ u_{21} & u_{22} & \cdots & u_{2j} \\ \vdots & \vdots & & \vdots \\ u_{j-1,1} & \cdot & \cdots & u_{j-1,j} \\ u_{i1} & \cdot & \cdots & u_{ij} \end{vmatrix} \div \begin{vmatrix} u_{11} & u_{12} & \cdots & u_{1j} \\ u_{21} & u_{22} & \cdots & u_{2j} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ u_{j1} & u_{j2} & \cdots & u_{jj} \end{vmatrix}. \tag{16}$$

Comparison with (10) shows that Λ_∞ is the lower triangle of the triangular decomposition $U_L U_R$ of the matrix U. Therefore, we have

$$A_\infty = \Lambda_\infty^{-1} A \Lambda_\infty = \Lambda_\infty^{-1} U \operatorname{diag}(\lambda_1, \ldots, \lambda_n)\, U^{-1} \Lambda_\infty = U_R \operatorname{diag}(\lambda_1, \ldots, \lambda_n)\, U_R^{-1}$$

$$= U_R \begin{pmatrix} \lambda_1 & & & & 0 \\ & \lambda_2 & & & \\ & & \cdot & & \\ & & & \cdot & \\ 0 & & & & \lambda_n \end{pmatrix} U_R^{-1} = \begin{pmatrix} \lambda_1 & * & * & * & * & * \\ & \lambda_2 & * & * & * & * \\ & & \cdot & * & * & * \\ & & & \cdot & * & * \\ & & & & \cdot & * \\ & & & & & \lambda_n \end{pmatrix}$$

because U_R is an upper (right) triangular matrix, which, applied as a transformation matrix upon diag $(\lambda_1, \ldots, \lambda_n)$, does not change the diagonal elements.

Although the conditions of theorem 3 are "practically always" fulfilled, there are very simple examples where lim Λ_k and lim A_k do not exist, for instance with

$$A = \begin{pmatrix} 1 & -1 & 1 \\ 4 & 6 & -1 \\ 4 & 4 & 1 \end{pmatrix},$$

where $\lambda_1 = 5$, $\lambda_2 = 2$, $\lambda_3 = 1$.

On the other hand, we have

THEOREM 4. *If the matrix A is hermitian and positive definite, then $\Lambda_\infty = \lim_{k\to\infty} \Lambda_k$, and $A_\infty = \lim_{k\to\infty} A_k$ exists and A_∞ is an upper triangular matrix.*

Note that the diagonal elements of A_∞ are the latent roots of A, but not necessarily ordered in absolute value.[9] Note further that theorem 4 does not cease to hold when A has multiple latent roots.

PROOF. Under the conditions stated in theorem 4 there exists a *unitary* matrix U so that

$$A = U \begin{pmatrix} \lambda_1 & & & 0 \\ & \cdot & & \\ & & \cdot & \\ 0 & & & \lambda_n \end{pmatrix} U^{-1}.$$

Therefore the matrix $V = U^{-1T}$ defined in (11) is U^C, the conjugate of U. In view of this, (13) reduces to

$$l_{ij}^{(k)} = \frac{\sum u^{(1)}_{\alpha_1\alpha_2 \ldots \alpha_j} \bar{u}_{\alpha_1\alpha_2 \ldots \alpha_j} (\lambda_{\alpha_1}\lambda_{\alpha_2} \cdots \lambda_{\alpha_j})^k}{\sum |u_{\alpha_1\alpha_2 \ldots \alpha_j}|^2 (\lambda_{\alpha_1}\lambda_{\alpha_2} \cdots \lambda_{\alpha_j})^k}, \tag{17}$$

where the sums have to be extended again over all possible combinations of j of the n values $1, 2, \ldots, n$. There is certainly one such combination $\alpha_1\alpha_2 \ldots \alpha_j$ with the properties

(a) $u_{\alpha_1\alpha_2 \ldots \alpha_j} \neq 0$ (because U is nonsingular),

(b) $\lambda_{\alpha_1}\lambda_{\alpha_2} \cdots \lambda_{\alpha_j} \geq \lambda_{\beta_1}\lambda_{\beta_2} \cdots \lambda_{\beta_j}$ if $u_{\beta_1\beta_2 \ldots \beta_j} \neq 0$.

Therefore the denominator in (17) will behave for large k like $c(\lambda_{\alpha_1}\lambda_{\alpha_2} \cdots \lambda_{\alpha_j})^k$ with $c \neq 0$. But the numerator cannot grow more rapidly (or decrease more slowly) since in the numerator any combination $\gamma_1\gamma_2 \ldots \gamma_j$ with $\lambda_{\gamma_1}\lambda_{\gamma_2} \cdots \lambda_{\gamma_j} > \lambda_{\alpha_1}\lambda_{\alpha_2} \cdots \lambda_{\alpha_j}$ is eliminated because, according to properties (a) and (b), $\bar{u}_{\gamma_1\gamma_2 \ldots \gamma_j} = 0$. Therefore $l_{ij}^{(k)}$ will converge for $k \to \infty$ and the existence of $\lim_{k\to\infty} A_k$ follows from theorem 2.

---

[9] This was overlooked by the author in the theorem stated in [7]. On the other hand, the restrictions imposed upon the latent vectors of A in that theorem are superfluous.

4. Numerical Examples

4.1. Wilson's Matrix (Turned Upside Down)

$$A=\begin{pmatrix} 10 & 9 & 7 & 5 \\ 9 & 10 & 8 & 6 \\ 7 & 8 & 10 & 7 \\ 5 & 6 & 7 & 5 \end{pmatrix}$$

$$L_1=\begin{pmatrix} 1. & 0 & 0 & 0 \\ 0.9 & 1. & 0 & 0 \\ 0.7 & 0.894737 & 1. & 0 \\ 0.5 & 0.789474 & 0.602941 & 1 \end{pmatrix}; \qquad R_1=\begin{pmatrix} 10 & 9. & 7. & 5. \\ & 1.9 & 1.7 & 1.5 \\ & & 3.578947 & 2.157895 \\ & & & 0.014706 \end{pmatrix}$$

$$A_2=\begin{pmatrix} 25.5 & 19.210529 & 10.014705 & 5. \\ 3.65 & 4.605264 & 2.604412 & 1.5 \\ 3.584210 & 4.905818 & 4.880030 & 2.157895 \\ 0.007353 & 0.011610 & 0.008867 & 0.014706 \end{pmatrix}.$$

Further calculation yields [10]

$$A_3=\begin{pmatrix} 29.658814 & 31.131274 & 10.019865 & 5. \\ 0.430404 & 3.249976 & 1.171746 & 0.784315 \\ 0.292581 & 2.474795 & 2.081053 & 0.522803 \\ 0.000003 & 0.000033 & 0.000010 & 0.010158 \end{pmatrix}$$

$$A_4=\begin{pmatrix} 30.209437 & 38.893433 & 10.019855 & 5. \\ 0.050732 & 3.593282 & 1.026337 & 0.711755 \\ 0.011711 & 0.919637 & 1.187133 & -0.077898 \\ 0 & 0 & 0 & 0.010149 \end{pmatrix}$$

$$A_5=\begin{pmatrix} 30.278627 & 41.462444 & 10.019855 & 5. \\ 0.006315 & 3.786811 & 1.009514 & 0.703360 \\ 0.000388 & 0.237012 & 0.924414 & -0.260174 \\ 0 & 0 & 0 & 0.010149 \end{pmatrix}$$

$$A_6=\begin{pmatrix} 30.287413 & 42.089697 & 10.019855 & 5. \\ 0.000802 & 3.841210 & 1.007420 & 0.702315 \\ 0.000010 & 0.053914 & 0.861228 & -0.304200 \\ 0 & 0 & 0 & 0.010149 \end{pmatrix}$$

with the corresponding transformation matrix

$$\Lambda_5=\begin{pmatrix} 1. & 0 & 0 & 0 \\ 1.059537 & 1. & 0 & 0 \\ 1.014577 & 3.177090 & 1. & 0 \\ 0.729909 & 2.170012 & 0.603971 & 1 \end{pmatrix}.$$

This example clearly shows the tendency of A_n for $k\rightarrow\infty$. See, however, section 6.

4.2. Matrix With Double Latent Roots

$$A_1=\begin{pmatrix} 6 & 4 & 4 & 1 \\ 4 & 6 & 1 & 4 \\ 4 & 1 & 6 & 4 \\ 1 & 4 & 4 & 6 \end{pmatrix}.$$

pos. def.

---

[10] The numbers given here are, of course, afflicted with roundoff errors.

Here we get

$$A_2=\begin{bmatrix} 11.5 & 3. & 6. & 1. \\ 1.666667 & 7.5 & 5. & 3.333333 \\ 2.5 & 3.75 & 12.5 & 5. \\ -1.25 & -7.5 & -15. & -7.5 \end{bmatrix}$$

$$A_3=\begin{bmatrix} 13.130434 & 4.615388 & 4.918032 & 1. \\ 1.575299 & 5.638798 & 0.680680 & 3.188405 \\ 1.672237 & 0.678113 & 5.722570 & 3.384615 \\ 0.053457 & 0.499369 & 0.532115 & -0.491803 \end{bmatrix}$$

$$A_4=\begin{bmatrix} 14.314568 & 4.797245 & 5.016867 & 1. \\ 0.634108 & 5.376677 & 0.393917 & 3.068432 \\ 0.661869 & 0.393172 & 5.411164 & 3.202761 \\ -0.004488 & -0.104187 & -0.108957 & -1.102410 \end{bmatrix}$$

$$A_5=\begin{bmatrix} 14.758727 & 4.943837 & 4.996771 & 1. \\ 0.235751 & 5.109737 & 0.110907 & 3.024134 \\ 0.238248 & 0.110906 & 5.112085 & 3.056174 \\ 0.000308 & 0.019496 & 0.019705 & -0.980550 \end{bmatrix}$$

$$\Lambda_4=\begin{bmatrix} 1. & 0 & 0 & 0 \\ 0.975866 & 1. & 0 & 0 \\ 0.975866 & -0.010594 & 1. & 0 \\ 0.951806 & 0.986213 & 0.996771 & 1 \end{bmatrix}.$$

In this example, the limits would be

$$A_\infty=\begin{bmatrix} 15 & 5 & 5 & 1 \\ 0 & 5 & 0 & 3 \\ 0 & 0 & 5 & 3 \\ 0 & 0 & 0 & -1 \end{bmatrix}, \qquad \Lambda_\infty=\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

4.3. Matrix With "Disorder of Latent Roots"

$$A_1=\begin{bmatrix} 5 & 4 & 1 & 1 \\ 4 & 5 & 1 & 1 \\ 1 & 1 & 4 & 2 \\ 1 & 1 & 2 & 4 \end{bmatrix}$$ — Hermitian positive definite

$$L_1=\begin{bmatrix} 1. & & & \\ 0.8 & 1 & & \\ 0.2 & 0.111111 & 1. & \\ 0.2 & 0.111111 & 0.470588 & 1 \end{bmatrix}; \qquad R_1=\begin{bmatrix} 5 & 4. & 1. & 1. \\ 0 & 1.8 & 0.2 & 0.2 \\ 0 & 0 & 3.777778 & 1.777778 \\ 0 & 0 & 0 & 2.941177 \end{bmatrix}$$

$$A_2=\begin{bmatrix} 8.6 & 4.222222 & 1.470588 & 1. \\ 1.52 & 1.844444 & 0.294118 & 0.2 \\ 1.111111 & 0.617284 & 4.614379 & 1.777778 \\ 0.588235 & 0.326797 & 1.384083 & 2.941177 \end{bmatrix}.$$

Further calculation yields:

$$A_3=\begin{bmatrix} 9.604650 & 4.352941 & 1.760563 & 1. \\ 0.200108 & 1.101232 & 0.040944 & 0.023256 \\ 0.683994 & 0.346020 & 4.899751 & 1.647059 \\ 0.163772 & 0.082852 & 0.694307 & 2.394367 \end{bmatrix}$$

and so on. The completion of the transformation of A_3 into triangular form is shown as an example for a faster converging method in section 6.

Note that in this example, the second diagonal element of A_k seems not to converge to the second largest latent root. The reason for this is that the condition (14b) of theorem 3 is violated. Indeed the matrix U, the columns of which are the latent vectors of A, is in this case

$$
\begin{array}{cccc} \lambda=10, & \lambda=5, & \lambda=2, & \lambda=1, \end{array}
$$

$$
U=\begin{pmatrix} 2 & 1 & 0 & 1 \\ 2 & 1 & 0 & -1 \\ 1 & -2 & 1 & 0 \\ 1 & -2 & -1 & 0 \end{pmatrix}
$$

so that

$$
\begin{vmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{vmatrix}=0.
$$

It can be shown that here

$$
A_\infty=\begin{pmatrix} 10 & 4.5 & 2 & 1 \\ 0 & 1. & 0 & 0 \\ 0 & 0 & 5 & 1.5 \\ 0 & 0 & 0 & 2. \end{pmatrix}; \qquad \Lambda_\infty=\begin{pmatrix} 1. & 0 & 0 & 0 \\ 1. & 1. & 0 & 0 \\ 0.5 & 0.25 & 1 & 0 \\ 0.5 & 0.25 & 1 & 1 \end{pmatrix}.
$$

In all these examples, we have convergence of A_k for $k\rightarrow\infty$, and the diagonal elements of A_∞ are the latent roots of A. However, it would not be a good practice to carry the LR-transformation so far that all subdiagonal elements of A_k are negligibly small, because the convergence of the LR-transformation is only linear. More exactly, the subdiagonal element of A_k in the ij position ($i>j$) converges to zero like $(\lambda_i/\lambda_j)^k$, provided (14) holds. Thus the convergence is poor if some of the latent roots are very close to each other, but even in the well-converging case 4.1, A_8 is still far from having negligibly small subdiagonal elements. For all these reasons a procedure with faster convergence will be developed in section 6 (see also sections 10 and 12).

5. The Case of a Real Matrix With Complex Latent Roots

Let A be a real nonsymmetric matrix with latent roots $\lambda_1,\lambda_2, \ldots \lambda_n$, where $\lambda_{m-1}=\bar{\lambda}_m$, but otherwise $|\lambda_i|>|\lambda_j|$ for $i<j$. Furthermore we require that (14b) be valid, i. e., that none of the "critical determinants"

$$
\begin{vmatrix} u_{11} & \cdots & u_{1j} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ u_{j1} & \cdots & u_{jj} \end{vmatrix} \quad \text{and} \quad \begin{vmatrix} v_{11} & \cdots & v_{1j} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ v_{j1} & \cdots & v_{jj} \end{vmatrix} \qquad (j=1, 2, \ldots n)
$$

vanish. With these assumptions, (13) still holds and $\lim\limits_{k\rightarrow\infty} l_{ij}^{(k)}$ exists for $j\neq m-1$, because then both numerator and denominator of $l_{ij}^{(k)}$ have exactly one dominating term, namely $u_{12\ldots j}^{(i)}v_{12\ldots j}(\lambda_1\lambda_2 \ldots \lambda_j)^k$ and $u_{12\ldots j}v_{12\ldots j}(\lambda_1\lambda_2 \ldots \lambda_j)^k$, respectively. Therefore (16) is still valid for $j\neq m-1$.

For $j=m-1$, however, we have two dominating terms, namely $u_{12\ldots m-1}^{(i)}v_{12\ldots m-1}(\lambda_1\lambda_2 \ldots \lambda_{m-1})^k$ and its conjugate in the numerator and two corresponding terms in the denominator.

Let $\varphi=\arg \lambda_{m-1}$, $\psi=\arg v_{12\ldots m-1}$. Then, for large k,

$$
l_{i,m-1}^{(k)}\sim\frac{\operatorname{Re}\{u_{12\ldots m-1}^{(i)}e^{i(k\varphi+\psi)}\}}{\operatorname{Re}\{u_{12\ldots m-1}e^{i(k\varphi+\psi)}\}}=\frac{a_k u_{12\ldots m-1}^{(i)}+\bar{a}_k\bar{u}_{12\ldots m-1}^{(i)}}{a_k u_{12\ldots m-1}+\bar{a}_k\bar{u}_{12\ldots m-1}}, \tag{18}
$$

which does not converge for $k\rightarrow\infty$.

It is shown in (18) that the $(m-1)$th column vector of Λ_k is asymptotically for large k a linear combination of two vectors, the components of which are $u^{(l)}_{12\ldots m-1}$ and $u^{(l)}_{12\ldots m-2,m}=\bar{u}^{(l)}_{12\ldots m-1}$, or

$$\begin{vmatrix} u_{11} & u_{12} & \ldots & u_{1m-1} \\ u_{21} & u_{22} & \ldots & u_{2m-1} \\ \vdots & \vdots & & \vdots \\ u_{m-2,1} & u_{m-2,2} & \ldots & u_{m-2,m-1} \\ u_{l1} & u_{l2} & \ldots & u_{l,m-1} \end{vmatrix} \quad \text{and} \quad \begin{vmatrix} u_{11} & u_{12} & \ldots & u_{1,m-2} & u_{1m} \\ u_{21} & u_{22} & \ldots & u_{2,m-2} & u_{2m} \\ \vdots & \vdots & & \vdots & \vdots \\ u_{m-2,1} & u_{m-2,2} & \ldots & u_{m-2,m-2} & u_{m-2,m} \\ u_{l1} & u_{l2} & \ldots & u_{l,m-2} & u_{lm} \end{vmatrix},$$

But the difference of two such column vectors for different values of k is parallel to one single vector, the components of which are

$$x_l=\begin{vmatrix} u_{12\ldots m-1} & u_{12\ldots m-2,m} \\ u^{(l)}_{12\ldots m-1} & u^{(l)}_{12\ldots m-2,m} \end{vmatrix}=u_{12\ldots m-2}\cdot u^{(l)}_{12\ldots m-1,m}, \tag{19}$$

(The term on the right side of (19) results from a well-known theorem of Sylvester (see Kowalewski [4], section 41). But as

$$\frac{u^{(l)}_{12\ldots m-1,m}}{u_{12\ldots m-1,m}}=\lim_{k\to\infty} l^{(k)}_{lm}$$

(see 16), we have

THEOREM 5. *Under the conditions stated at the beginning of section 5, all column vectors of* Λ_k *converge for* $k\to\infty$, *except the* $(m-1)$*th column, which changes from* k *to* $k+1$ *asymptotically by a multiple of the* m*th column vector of* Λ_k.

From theorem 5, and the relation $\Lambda_k=\Lambda_{k-1}L_k$ which follows from (6), we infer that asymptotically for large k:

$$L_k=\begin{bmatrix} 1 & & & & & & & \\ & 1 & & & & \text{O} & & \\ & & 1 & & & & & \\ & & & 1 & & & & \\ & & & x & 1 & & & \\ & & & & & 1 & & \\ & & & & & & 1 & \\ & & & & & & & 1 \end{bmatrix} \tag{20}$$

where x (in the mth row and the $(m-1)$th column) is the only off-diagonal element of L_k that does not converge to zero for $k\to\infty$.

But as $\Lambda_k^{-1}=L_k^{-1}\Lambda_{k-1}^{-1}$, all row vectors of Λ_k^{-1} converge for $k\to\infty$, except the mth-row vector which changes from k to $k+1$ asymptotically by a multiple of the $(m-1)$th row. Therefore $A_k=\Lambda_{k-1}^{-1}A\Lambda_{k-1}$ converges for $k\to\infty$ except the mth-row vector and the $(m-1)$th column vector. But as L_k, the lower triangle of the triangular decomposition of A_k, has for large k the form (20), we conclude that either (a) all elements of A_k below the diagonal converge to zero for $k\to\infty$, except the element in the same position as the x in L_k (see 20), or (b) some of the diagonal elements of A_k tend to infinity.

Now, for some k, the denominator in (18) may be very small, and therefore Λ_k as well as A_{k+1} may indeed have some very large elements. There is an infinity of such values of k, but for all other k's the elements of Λ_k and A_{k+1} remain in the same range of size. Therefore (b) cannot be true and we have:

THEOREM 6. *Let* A *be a real matrix with latent roots* $\lambda_1\lambda_2\ldots\lambda_n$ *and let—as in section 3—the columns of the matrices* U *and* V *be the corresponding latent vectors of* A *and* A^T.

Furthermore, let the latent roots of A *be ordered in absolute value:* $|\lambda_i|\geq|\lambda_k|$ *for* $i<k$, *and denote with*

λ_r *any real latent root, with* λ_{q-1}, λ_q *any pair of conjugate complex roots. If further*

$$|\lambda_r| \neq |\lambda_{r'}| \quad \text{for} \quad r \neq r'$$

$$|\lambda_q| \neq |\lambda_{q'}| \quad \text{for} \quad q \neq q'$$

$$|\lambda_r| \neq |\lambda_q| \quad \text{for any } r,\ q$$

and

$$\left|\begin{array}{ccc|ccc} u_{11} & \cdots & u_{1j} & v_{11} & \cdots & v_{1j} \\ \cdot & & \cdot & \cdot & & \cdot \\ \cdot & & \cdot & \cdot & & \cdot \\ \cdot & & \cdot & \cdot & & \cdot \\ u_{j1} & \cdots & u_{jj} & v_{j1} & \cdots & v_{jj} \end{array}\right| \neq 0 \quad \text{for} \quad j=1,2,\ \ldots\ n,$$

then

(a) *the subdiagonal elements of* A_k *converge to zero for* $k \to \infty$, *except the elements* $a_{q,q-1}$,

(b) *the rth diagonal element of* A_k *converges to* λ_r,

(c) *except for the elements in the qth row and/or in the* $(q-1)$*th column all elements of* A_k *above the diagonal converge for* $k \to \infty$,

(d) *the two-row minors*

$$M_q = \begin{vmatrix} a_{q-1,\,q-1} & a_{q-1,\,q} \\ a_{q,\,q-1} & a_{q,\,q} \end{vmatrix}$$

of A_k *do not converge for* $k \to \infty$, *but the latent roots of* M_q *converge to* λ_{q-1} *and* λ_q,

(e) *except for the* $(q-1)$*th column, all columns of the transformation matrix* Λ_k *converge for* $k \to \infty$ *and the limits are given by* (16).

Numerical example. For

$$A_1 = \begin{pmatrix} 4 & -5 & 0 & 3 \\ 0 & 4 & -3 & -5 \\ 5 & -3 & 4 & 0 \\ 3 & 0 & 5 & 4 \end{pmatrix},$$

we obtain

$$A_2 = \begin{pmatrix} 6.25 & -2.1875 & 3.64078 & 3. \\ -7.5 & -3.125 & -9.06796 & -5. \\ 8.28125 & 5.52344 & 6.81675 & 0.3125 \\ 4.54369 & 5.67961 & 7.35225 & 6.05825 \end{pmatrix}$$

$$A_3 = \begin{pmatrix} 15.88000 & -11.31304 & 4.39889 & 3. \\ -0.34400 & 2.90261 & -5.05282 & -1.4 \\ -10.63235 & 14.38518 & -6.33355 & -5.71304 \\ 2.58150 & -4.48956 & 0.89734 & 3.55094 \end{pmatrix}$$

$$A_4 = \begin{pmatrix} 13.66750 & -3.03182 & 2.86547 & 3. \\ 3.04469 & -8.71587 & -4.27515 & -1.33501 \\ -6.28388 & 24.15839 & 9.46129 & -0.28312 \\ 0.25800 & -1.58285 & -0.81122 & 1.58707 \end{pmatrix}$$

$$A_5 = \begin{pmatrix} 11.73129 & -10.57540 & 2.80131 & 3. \\ 0.43008 & 5.49061 & -4.87064 & -2.00332 \\ 1.35383 & 8.00050 & -3.03460 & -4.57567 \\ 0.03422 & 0.34395 & -0.03877 & 1.81271 \end{pmatrix}$$

$$\Lambda_4 = \begin{pmatrix} 1. & 0 & 0 & 0 \\ -0.99889 & 1. & 0 & 0 \\ 0.91043 & -0.92066 & 1. & 0 \\ 0.91227 & -1.85846 & 0.93377 & 1 \end{pmatrix}$$

Already, A_5 gives some indications about the latent roots, which in this case are 12, $1\pm 5i$, and 2. However, we are still far from a matrix with small subdiagonal elements. Therefore we shall develop in section 6 also a method to speed up the convergence in such cases where A has conjugate complex latent roots.

It should be noted that the behavior of A_k is nearly the same if A has, instead of two conjugate complex latent roots, two real ones of equal or nearly equal absolute value.

6. Improvement of Convergence[11]

As already pointed out in section 4, it would take too long a time to complete the transformation to triangular form with the LR transformation alone. The LR-transformation is certainly a useful tool for the first steps, until the diagonal elements of A_k are ordered in absolute value and the subdiagonal elements show a definite tendency to converge to zero. But for the later stages we advocate a slightly different procedure, which can be described as follows:

6.1. The Case of Real Latent Roots

Let A be a matrix for which (14) holds, and let A_m be the matrix obtained from $A=A_1$ by $m-1$ single LR-steps and Λ_{m-1} the corresponding transformation matrix with the property $A_m=\Lambda_{m-1}^{-1}A_1\Lambda_{m-1}$. Then we carry out a transformation,

$$\begin{aligned} A_{m+1}&=L_m^{-1}A_mL_m \\ \Lambda_m&=\Lambda_{m-1}L_m, \end{aligned} \tag{21}$$

where L_m is a matrix of the following kind:

$$L_m=\begin{bmatrix} 1 & & & & \\ x_2 & 1 & & \mathbf{0} & \\ x_3 & & 1 & & \\ \vdots & \mathbf{0} & & \ddots & \\ x_n & & & & 1 \end{bmatrix};\quad L_m^{-1}=\begin{bmatrix} 1 & & & & \\ -x_2 & 1 & & \mathbf{0} & \\ -x_3 & & 1 & & \\ \vdots & \mathbf{0} & & \ddots & \\ -x_n & & & & 1 \end{bmatrix}. \tag{22}$$

By proper choice of the x's we could succeed in making all subdiagonal elements of the first column of A_{m+1} exactly zero. However, this would require the determination of a latent vector of A_m. Without an undue amount of work we cannot make these elements exactly zero, but only very small. Let us express these elements explicitly in terms of A_m and the x's.

If we denote the elements of A_m transitorily by a_{ij}, and those of A_{m+1} by b_{ij}, then it follows from (21) that

$$b_{j1}=a_{j1}-a_{11}x_j+\sum_2^n a_{jr}x_r-x_j\sum_2^n a_{1r}x_r \qquad (j=2,\,3,\,\ldots\,,\,n). \tag{23}$$

By neglecting the quadratic terms in x and the subdiagonal elements of the other columns of A_m (these are the elements a_{ij}, with $1<j<i\leq n$), we get the following linear equations for the x's:

$$\sum_{r=j}^n (a_{jr}-\delta_{jr}a_{11})x_r+a_{j1}=0 \qquad (j=2,\,3,\,\ldots\,,\,n), \tag{24}$$

---

[11] For another method with improved (quadratic) convergence see section 12; and for a further method for striped matrices see section 1

or explicitly,

$$\begin{array}{cccccc}
x_2 & x_3 & x_4 & \cdots & x_n & 1 \quad = 0 \\
\hline
a_{22}-a_{11} & a_{23} & a_{24} & \cdots & a_{2n} & a_{21} \\
 & a_{33}-a_{11} & a_{34} & & \vdots & \vdots \\
 & & \ddots & & & \\
 & & & & a_{nn}-a_{11} & a_{n1}
\end{array} \tag{25}$$

If we solve for the x's and execute the transformation $L_m^{-1}A_mL_m=A_{m+1}$, the subdiagonal elements of the first column of A_{m+1} will not be exactly zero but will be considerably smaller than in A_m.

In the next step we try to eliminate the subdiagonal elements of the second column; this is accomplished by transformation with a transformation matrix

$$L_{m+1}=\begin{bmatrix}
1 & & & & \\
0 & 1 & & \mathbf{O} & \\
0 & y_3 & 1 & & \\
0 & y_4 & & 1 & \\
\vdots & \vdots & \mathbf{O} & & \ddots \\
0 & y_n & & & & 1
\end{bmatrix}; \quad
L_{m+1}^{-1}=\begin{bmatrix}
1 & & & & \\
0 & 1 & & \mathbf{O} & \\
0 & -y_3 & 1 & & \\
0 & -y_4 & & 1 & \\
\vdots & \vdots & \mathbf{O} & & \ddots \\
0 & -y_n & & & & 1
\end{bmatrix}, \tag{26}$$

where the y's are determined by the equations

$$\begin{array}{ccccc}
y_3 & y_4 & \cdots & y_n & 1 \quad = 0 \\
\hline
a_{33}-a_{22} & a_{34} & \cdots & a_{3n} & a_{32} \\
 & a_{44}-a_{22} & \cdots & a_{4n} & a_{42} \\
 & & \ddots & \vdots & \vdots \\
 & & & a_{nn}-a_{22} & a_{n2}
\end{array} \tag{27}$$

In this way we work through all columns, i. e., we apply in sequence the following transformations;

$$\begin{bmatrix}
1 & & & \\
x_2 & 1 & \mathbf{O} & \\
x_3 & & 1 & \\
\vdots & \mathbf{O} & & \ddots \\
x_n & & & & 1
\end{bmatrix}; \quad
\begin{bmatrix}
1 & & & & \\
0 & 1 & & \mathbf{O} & \\
0 & y_3 & 1 & & \\
0 & y_4 & & 1 & \\
\vdots & \vdots & \mathbf{O} & & \ddots \\
0 & y_n & & & & 1
\end{bmatrix}; \dots ; \quad
\begin{bmatrix}
1 & & & & \\
 & 1 & & \mathbf{O} & \\
 & & 1 & & \\
 & & & \ddots & \\
 & \mathbf{O} & & & 1 & \\
 & & & & z_n & 1
\end{bmatrix}, \tag{28}$$

where the x's, y's, . . . , z's are determined by linear equations of the type (25), (27). In the sequel, the application of the $n-1$ transformations (28) will be called a "sweep."

After the completion of the first sweep, a new matrix A_{m+n-1} has been computed, and we compute also the corresponding transformation matrix[12]

$$\Lambda_{m+n-2}=\Lambda_{m-1}.L_mL_{m+1}\;.\;.\;.\;L_{m+n-2}. \tag{29}$$

Then a second sweep is started, and so on, until all subdiagonal elements are negligibly small. This will happen very soon, because the procedure has *quadratic convergence;* i. e., if once all subdiagonal elements are small (of order ϵ) then one further sweep will make them of order $c\cdot\epsilon^2$, where c depends only upon the matrix A.

Numerical example. We pick up the matrix A_1 of section 4.3, which has already been transformed into

$$A_3=\begin{pmatrix} 9.604650 & 4.352941 & 1.760563 & 1. \\ 0.200108 & 1.101232 & 0.040944 & 0.023256 \\ 0.683994 & 0.346020 & 4.899751 & 1.647059 \\ 0.163772 & 0.082852 & 0.694307 & 2.394367 \end{pmatrix}$$

with

$$\Lambda_2=\begin{pmatrix} 1. & 0 & 0 & 0 \\ 0.976744 & 1. & 0 & 0 \\ 0.348837 & 0.176470 & 1. & 0 \\ 0.348837 & 0.176470 & 0.760563 & 1 \end{pmatrix}.$$

From (25) we obtain the following equations for the x's:

	x_2	x_3	x_4	1
	−8.503418	0.040944	0.023256	0.200108
		−4.704899	1.647059	0.683994
			−7.210283	0.163772
Solution:	0.024333	0.153331	0.022714	

Therefore the first transformation matrix is

$$L_3=\begin{pmatrix} 1. & 0 & 0 & 0 \\ 0.024333 & 1 & 0 & 0 \\ 0.153331 & 0 & 1 & 0 \\ 0.022714 & 0 & 0 & 1 \end{pmatrix};$$

and

$$A_4=L_3^{-1}A_3L_3=\begin{pmatrix} 10.003233 & 4.352941 & 1.760563 & 1. \\ -0.009698 & 0.995312 & -0.001896 & -0.001077 \\ -0.052698 & -0.321421 & 4.629802 & 1.493728 \\ 0.099419 & -0.016021 & 0.654318 & 2.371653 \end{pmatrix}.$$

Here the subdiagonal elements of the first column are considerably smaller than in A_3, only the element a_{14} did not improve so much. The latter is due to the fact that in setting up the equations for the x's, the comparatively large element $a_{43}=0.694307$ of A_3 has been neglected. One might ask, therefore, whether it would not be an advantage to neglect only the quadratic terms in x and solve the full linear system in the x's. This, however, would increase computational labor enormously for large n.

---

[12] This is used for the determination of the latent-vectors (see also section 7), as well as for checking purposes (see section 8).

Now A_4 is transformed again in order to eliminate the subdiagonal elements of the second column. The transformation matrix L_4 is given by the equations (27).

$$L_4=\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1. & 0 & 0 \\ 0 & 0.083652 & 1 & 0 \\ 0 & 0.011640 & 0 & 1 \end{pmatrix}$$

$$A_5=L_4^{-1}A_4L_4=\begin{pmatrix} 10.003233 & 4.511856 & 1.760563 & 1. \\ -0.009698 & 0.995141 & -0.001896 & -0.001077 \\ -0.051887 & 0.000013 & 4.269961 & 1.493818 \\ 0.099532 & 0.054737 & 0.654340 & 2.371666 \end{pmatrix}.$$

The next step, which is a transformation by the matrix

$$L_5=\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1. & 0 \\ 0 & 0 & 0.289750 & 1 \end{pmatrix},$$

results in

$$A_6=\begin{pmatrix} 10.003233 & 4.511856 & 2.050313 & 1. \\ -0.009698 & 0.995141 & -0.002208 & -0.001077 \\ -0.051887 & 0.000013 & 5.062795 & 1.493818 \\ 0.114566 & 0.054733 & -0.125414 & 1.938832 \end{pmatrix}$$

and

$$\Lambda_5=\begin{pmatrix} 1. & 0 & 0 & 0 \\ 1.001077 & 1. & 0 & 0 \\ 0.506462 & 0.260122 & 1. & 0 \\ 0.492463 & 0.251733 & 1.050313 & 1 \end{pmatrix}.$$

This completes the first sweep. After the second one, which consists of the three transformations

$$L_6=\begin{pmatrix} 1. & 0 & 0 & 0 \\ -0.001077 & 1 & 0 & 0 \\ -0.006207 & 0 & 1 & 0 \\ -0.014206 & 0 & 0 & 1 \end{pmatrix}, \quad L_7=\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1. & 0 & 0 \\ 0 & -0.010601 & 1 & 0 \\ 0 & 0.010125 & 0 & 1 \end{pmatrix}, \quad L_8=\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1. & 0 \\ 0 & 0 & -0.049047 & 1 \end{pmatrix},$$

we have

$$A_9=\begin{pmatrix} 9.999853 & 4.500246 & 2.001266 & 1. \\ -0.000002 & 1. & 0 & 0 \\ -0.000022 & 0 & 5.001949 & 1.500025 \\ 0.000770 & 0.001638 & -0.003608 & 1.998198 \end{pmatrix}$$

$$\Lambda_8=\begin{pmatrix} 1. & 0 & 0 & 0 \\ 1. & 1. & 0 & 0 \\ 0.499975 & 0.249521 & 1. & 0 \\ 0.499897 & 0.250724 & 1.001266 & 1 \end{pmatrix}$$

and after the third sweep:

$$A_{12}=\begin{pmatrix} 9.999997 & 4.499999 & 2.000001 & 1. \\ -0.000002 & 1. & 0 & 0 \\ +0.000002 & 0.000001 & 5.000003 & 1.500001 \\ 0.000001 & -0.000002 & -0.000002 & 2. \end{pmatrix}$$

$$\Lambda_{11}=\begin{pmatrix} 1. & 0 & 0 & 0 \\ 1. & 1. & 0 & 0 \\ 0.499999 & 0.250001 & 1. & 0 \\ 0.499999 & 0.249997 & 1.000001 & 1 \end{pmatrix}.$$

It should be noted that the amount of work for one sweep is about the same as for one single LR-transformation sweep.

6.2. The Case of Conjugate Complex Latent Roots

Let A be a matrix that fulfills the conditions of theorem 6 and let the latent roots be ordered in absolute value. We shall use here the same notations; λ_r shall denote a real root, λ_{q-1}, λ_q a complex pair. Then, if the matrix A has been transformed by $m-1$ LR-transformation steps into a matrix A_m, we try to make the subdiagonal elements of A_m columnwise to zero by transforming it in a similar manner as in section 6.1.

Let r be a column whose diagonal element a_{rr} corresponds to a real latent root. Then we choose a transformation matrix

$$L=\begin{pmatrix} 1 &&&&&&& \\ & 1 &&&&&& \\ && \ddots &&& \mathbf{O} && \\ &&& 1 &&&& \\ &&& x_{r+1} & 1 &&& \\ & \mathbf{O} && \vdots && 1 && \\ &&& \vdots &&& 1 & \\ &&& \vdots && \mathbf{O} && \ddots & \\ &&& x_n &&&&& 1 \end{pmatrix}, \tag{30}$$

where the x's are determined by the same kind of equations as (25) or (27), except that there are subdiagonal elements $a_{q,q-1}$ which can not be neglected because they are not small.

If after some steps of the kind mentioned above we come to a pair of columns $q-1,q$ corresponding to a conjugate complex pair of latent roots, then the two columns have to be treated simultaneously by a transformation matrix of the following kind:

$$L=\begin{pmatrix} 1 &&&&&& \\ & \ddots &&&&& \\ && 1 &&& \mathbf{O} & \\ && 0 & 1 &&& \\ && x_{q+1} & y_{q+1} & 1 && \\ & \mathbf{O} & \vdots & \vdots && 1 & \\ && \vdots & \vdots && \mathbf{O} & \ddots & \\ && x_n & y_n &&&& 1 \end{pmatrix}. \tag{31}$$

If a_{ij} are the elements before, b_{ij} after, the transformation $L^{-1}AL$ with the matrix (31), then we have for the elements of the two columns $q-1$ and q (for $i=q+1, q+2, \ldots, n$).

$$b_{i,q-1}=a_{i,q-1}+\sum_{q+1}^{n} a_{ij}x_j-a_{q-1,q-1}x_i-a_{q,q-1}y_i+\text{quadratic terms in } x,y.$$

$$b_{iq}=a_{iq}+\sum_{q+1}^{n} a_{ip}y_p-a_{q-1,q}x_i-a_{qq}y_i+\text{quadratic terms in } x,y.$$

This gives us the following equations for the x,y if we want to eliminate approximately the subdiagonal elements of the columns $q-1,q$:

$$\left.\begin{aligned} &\sum_{q+1}^{n} a_{ip}x_p-a_{q-1,q-1}x_i-a_{q,q-1}y_i=0 \\ &\sum_{q+1}^{n} a_{ip}y_p-a_{q-1,q}x_i-a_{qq}y_i=0 \end{aligned}\right\} \quad \text{for} \quad i=q+1, \ldots, n. \tag{32}$$

Here all the elements a_{ij} with $j<i$ may be neglected, unless $j=q'-1$; $i=q'$ where $\lambda_{q'-1},\lambda_{q'}$ is another conjugate complex pair of latent roots.

An example may illustrate the nature of eq (32). Let $\lambda_1, \ldots, \lambda_6$ be the latent roots of a six-row matrix and let λ_1 and λ_4 be real roots, $\lambda_2\lambda_3$ and $\lambda_5\lambda_6$ two pairs of complex roots. Furthermore, $|\lambda_1|>|\lambda_2|>|\lambda_4|>|\lambda_5|$.

Then we have the following equations:

1. For the elements of the matrix (30) with $r=1$:

$$\begin{array}{cccccccc}
x_2 & x_3 & x_4 & x_5 & x_6 & 1 & = & 0 \\
\hline
a_{22}-a_{11} & a_{23} & a_{24} & a_{25} & a_{26} & a_{21} & & \\
a_{32} & a_{33}-a_{11} & a_{34} & a_{35} & a_{36} & a_{31} & & \\
 & & a_{44}-a_{11} & a_{45} & a_{46} & a_{41} & & \\
 & & & a_{55}-a_{11} & a_{56} & a_{51} & & \\
 & & & a_{65} & a_{66}-a_{11} & a_{61} & &
\end{array}$$

2. For the elements of the matrix (31) with $q-1,q=2,3$:

$$\begin{array}{ccccccccc}
x_4 & y_4 & x_5 & y_5 & x_6 & y_6 & 1 & = & 0 \\
\hline
a_{44}-a_{22} & -a_{32} & a_{45} & 0 & a_{46} & 0 & a_{42} & & \\
-a_{23} & a_{44}-a_{33} & 0 & a_{45} & 0 & a_{46} & a_{43} & & \\
 & & a_{55}-a_{22} & -a_{32} & a_{56} & 0 & a_{52} & & \\
 & & -a_{23} & a_{55}-a_{33} & 0 & a_{56} & a_{53} & & \\
 & & a_{65} & 0 & a_{66}-a_{22} & -a_{32} & a_{62} & & \\
 & & 0 & a_{65} & -a_{23} & a_{66}-a_{33} & a_{63} & &
\end{array}$$

Numerical example. We pick up the matrix A of the example in section 5 which has already been transformed into

$$A_5=\begin{pmatrix}
11.70129 & -10.57540 & 2.80131 & 3. \\
0.43008 & 5.49061 & -4.87064 & -2.00332 \\
1.35383 & 8.00050 & -3.03460 & -4.57567 \\
0.03422 & 0.34395 & -0.03877 & 1.81271
\end{pmatrix},$$

with

$$\Lambda_4=\begin{pmatrix}
1. & 0 & 0 & 0 \\
-0.99889 & 1. & 0 & 0 \\
0.91043 & -0.92066 & 1. & 0 \\
0.91227 & -1.85846 & 0.93372 & 1
\end{pmatrix}.$$

Here columns 1 and 4 seem to correspond to real latent roots, $q-1=2$ and $q=3$ to a conjugate complex pair. To eliminate the subdiagonal elements of the first column we apply a transformation of type (30) If the x's are determined by eq (27), we obtain

$$L_5=\begin{pmatrix}
1. & 0 & 0 & 0 \\
-0.00205 & 1 & 0 & 0 \\
0.08951 & 0 & 1 & 0 \\
0.00345 & 0 & 0 & 1
\end{pmatrix}$$

and

$$A_6=L_5^{-1}A_5L_5=\begin{bmatrix} 12.01406 & -10.57540 & 2.80131 & 3. \\ 0.00057 & 5.46893 & -4.86490 & -1.99717 \\ -0.02537 & 8.94710 & -3.28535 & -4.84420 \\ -0.00515 & 0.38044 & -0.04843 & 1.80236 \end{bmatrix}.$$

Now we come to the treatment of the second and third columns by a transformation of type (31). In this case eqs (32) degenerate to two equations for x_4 and y_4, from which we obtain

$$L_6=\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1. & 0 & 0 \\ 0 & 0 & 1. & 0 \\ 0 & -0.06040 & 0.06727 & 1 \end{bmatrix}$$

$$A_7=\begin{bmatrix} 12.01406 & -10.75660 & 3.00312 & 3. \\ 0.00057 & 5.58956 & -4.99925 & -1.99717 \\ -0.02537 & 9.23969 & -3.61122 & -4.84420 \\ -0.00341 & -0.01237 & 0.01379 & 2.00760 \end{bmatrix}$$

$$\Lambda_6=\begin{bmatrix} 1. & 0 & 0 & 0 \\ -1.00094 & 1. & 0 & 0 \\ 1.00183 & -0.92066 & 1. & 0 \\ 1.00311 & -1.91886 & 1.00099 & 1 \end{bmatrix}.$$

This completes one sweep; the next sweep gives the following result.

$$A_9=\begin{bmatrix} 12.00005 & -10.76197 & 3.00000 & 3. \\ 0.00002 & 5.60325 & -4.99999 & -1.99999 \\ -0.00005 & 9.23803 & -3.60331 & -4.84132 \\ -0.00003 & 0 & 0.00002 & 2.00002 \end{bmatrix}$$

$$\Lambda_8=\begin{bmatrix} 1. & 0 & 0 & 0 \\ -1. & 1. & 0 & 0 \\ 1. & -0.92066 & 1. & 0 \\ 1.00001 & -1.92065 & 0.99995 & 1 \end{bmatrix}.$$

From A_9 we may read the latent roots, which are 12 and 2, together with the latent roots of the minor

$$\begin{pmatrix} 5.60325 & -4.99999 \\ 9.23803 & -3.60331 \end{pmatrix},$$

which are $0.99997 \pm 4.99998i$.

6.3. Special Remarks

As in the example of section 4.2, it may happen that two diagonal elements of A_m are nearly equal. This may lead to large elements in some of the transformation matrices in (28) so that the quadratic terms in (23) are no longer negligible. In such cases it may be worthwhile to use a transformation matrix with only one nonzero off-diagonal element, but this one chosen so that one subdiagonal element vanishes exactly.[13]

---

[13] This is essentially Jacobi's idea; see Jacobi [3 and Greenstadt [15].

Let us consider the four elements $a_{ii}a_{ij}a_{ji}a_{jj}$ of the matrix A_m. Then the transformation with the matrix

$$L_m=\begin{pmatrix} 1 & & & & & & & \\ & 1 & & & & & & \\ & & \cdot & & & & \mathbf{O} & \\ & & & \cdot & & & & \\ & & & & \cdot & & & \\ & & & & & \cdot & & \\ & x & & & & & 1 & \\ & & & & & & & 1 \end{pmatrix},$$

where x, in the ith column and jth row $(i<j)$, is the only nonzero off-diagonal element, transforms A_m so that the four elements at the crosspoints of the rows and columns i and j become

$$\begin{array}{ccccc} & \vdots & & \vdots & \\ \cdots\cdots & a_{ii}+xa_{ij} & \cdots\cdots\cdots\cdots & a_{ij} & \cdots\cdots \\ & \vdots & & \vdots & \\ \cdots\cdots & a_{ji}-xa_{ii}+xa_{jj}-x^2a_{ij} & \cdots\cdots\cdots\cdots & a_{jj}-xa_{ij} & \cdots\cdots \\ & \vdots & & \vdots & \end{array} \qquad (33)$$

To eliminate the lower left of these four elements, we have to choose x as one solution of the equation

$$-a_{ij}x^2+(a_{jj}-a_{ii})x+a_{ji}=0 \qquad (34)$$

and to execute the transformation $L_m^{-1}A_mL_m$. Then we continue as usual.

This trick may be useful also in other cases where trouble occurs. Take, for instance, the matrix

$$A=\begin{pmatrix} 2 & 1 & 3 & 4 \\ 1 & -3 & 1 & 5 \\ 3 & 1 & 6 & -2 \\ 4 & 5 & -2 & -1 \end{pmatrix},$$

which is used by Bodewig [11] for criticism [14] against the power method [12]. Here the convergence of the LR-transformation is poor because two latent roots are very close in absolute value and opposite in sign. Indeed, we have after seven single LR steps:

$$A_8=\begin{pmatrix} 4.82840 & -7.50426 & -0.93851 & 4. \\ -5.20416 & -4.83364 & -3.17454 & 1.80945 \\ -0.11019 & -0.53746 & 5.57692 & -3.78443 \\ 0.00034 & 0.00022 & -0.00277 & -1.57168 \end{pmatrix}$$

$$\Lambda_7=\begin{pmatrix} 1. & 0 & 0 & 0 \\ 0.79764 & 1. & 0 & 0 \\ 0.90851 & -1.02218 & 1. & 0 \\ -0.17369 & -1.35943 & -0.98463 & 1 \end{pmatrix},$$

with no indication of convergence in the first two rows of A_k. This suggests the application of the methods given in section 6.2; in this way we could get two roots directly and the two other roots as latent

---

[14] The criticism of E. Bodewig is wholly unjustified because the two latent roots in question ($\lambda_1=-8.028\ldots$ and $\lambda_2=+7.932\ldots$) behave and may be treated like a conjugate complex pair. See also Wilkinson [17].

roots of a two-row minor. In this case, however, we may eliminate the worst of the subdiagonal elements, $a_{21}=5.20416$, by the trick mentioned above: with $i=1, j=2$ we obtain the equation

$$5.20416-9.66204x+7.50426x^2=0,$$

thus yielding

$$L_8=\begin{pmatrix} 1. & 0 & 0 & 0 \\ -0.40881 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$L_8^{-1}A_8L_8=\begin{pmatrix} 7.89622 & -7.50426 & -0.93851 & 4. \\ -0.00006 & -7.90146 & -3.55821 & 3.44469 \\ 0.10953 & -0.53746 & 5.57692 & -3.78443 \\ 0.00025 & 0.00022 & -0.00277 & -1.57168 \end{pmatrix}.$$

Two sweeps will turn this into

$$A_\infty=\begin{pmatrix} 7.93298 & -7.51608 & -0.94011 & 4. \\ -0.00002 & -8.02866 & -3.56995 & 3.48877 \\ -0.00001 & 0.00002 & 5.66886 & -4.02376 \\ 0.00001 & 0 & 0.00001 & -1.57319 \end{pmatrix}$$

$$\Lambda_\infty=\begin{pmatrix} 1. & 0 & 0 & 0 \\ 0.37781 & 1. & 0 & 0 \\ 1.38664 & -1.00975 & 1. & 0 \\ 0.34881 & -1.37171 & -0.98503 & 1 \end{pmatrix}.$$

7. Determination of Latent Vectors

If A_∞ and Λ_∞ have been determined, then the computation of the latent vectors of A is trivial: Because $A_\infty=\Lambda_\infty^{-1}A\Lambda_\infty$, we need only to compute the latent vectors y of A_∞. Then $v=\Lambda_\infty y$ are the latent vectors of A. But as A_∞ is triangular, the determination of its latent vectors is trivial.

Numerical example. For Bodewig's matrix

$$\begin{pmatrix} 2 & 1 & 3 & 4 \\ 1 & -3 & 1 & 5 \\ 3 & 1 & 6 & -2 \\ 4 & 5 & -2 & -1 \end{pmatrix}$$

we have approximately

$$A_\infty=\begin{pmatrix} 7.93298 & -7.51608 & -0.94011 & 4. \\ 0 & -8.02866 & -3.56995 & 3.48877 \\ 0 & 0 & 5.66886 & -4.02376 \\ 0 & 0 & 0 & -1.57319 \end{pmatrix},$$

y_1 y_2 y_3 y_3

and therefore the latent vector of A_∞ belonging to $\lambda=5.66886$ (as an example) is determined by the equations:

$$\begin{array}{lcccc} & y_1 & y_2 & y_3 & y_4 \\ \hline & 2.26412 & -7.51608 & -0.94011 & 4. \quad =0 \\ & 0 & -13.69752 & -3.56995 & 3.48877=0 \\ & 0 & 0 & 0 & -4.02376=0 \\ & 0 & 0 & 0 & -7.24205=0 \\ \hline \text{Solution:} & -0.44998 & -0.26063 & 1 & 0 \end{array}$$

This gives

$$y=(-0.44898,\quad -0.26063,\quad 1,\quad 0)^T,$$

$$v=\Lambda_\infty y=(-0.44898,\quad -0.43064,\quad 0.63921,\quad -0.78448)^T.$$

As a check we compute $A(\Lambda_\infty y)/5.66886$ and obtain

$$(-0.448983,\quad -0.430642,\quad 0.639218,\quad -0.784472)^T.$$

The case of complex conjugate latent roots provides some more difficulties insofar as one has to compute in the complex domain and the matrix A is not strictly triangular.

8. Corrective Measures Against Roundoff Errors, Estimates

It is clear that the transformation to triangular form, if carried out numerically, can only be approximate. Therefore, the accuracy of A_∞ should always be checked. In the sequel, the approximate triangular form and the corresponding transformation matrix, as computed by the methods given in sections 1 to 6, shall be denoted by $A_{\overline{\infty}}$ and $\Lambda_{\overline{\infty}}$.

The easiest check is based upon the relation $A_\infty=\Lambda_\infty^{-1}A\Lambda_\infty$; therefore we compute

$$\Lambda_{\overline{\infty}}^{-1}A\Lambda_{\overline{\infty}} \tag{35}$$

in one step and (if needed) with higher accuracy. The result of this transformation will be somewhat different from $A_{\overline{\infty}}$ and especially the subdiagonal elements may not be as small as expected. However, the result of (35) is more accurate as far as similarity to the original matrix A is concerned, because less computation is involved. Therefore $A_{\overline{\infty}}$ is disposed of and the result of (35) is used instead to calculate a better approximation to A_∞; this can be done by some further sweeps of the kind mentioned in section 6.

Numerical example. For Bodewig's matrix we found in section 6:

$$\Lambda_{\overline{\infty}}=\begin{pmatrix} 1. & 0 & 0 & 0 \\ 0.37781 & 1. & 0 & 0 \\ 1.38664 & -1.00975 & 1. & 0 \\ 0.34881 & -1.37171 & -0.98503 & 1 \end{pmatrix}.$$

If we compute $\Lambda_{\overline{\infty}}^{-1}A\Lambda_{\overline{\infty}}$ with eight digits after the decimal point, we obtain the following matrix in place of the $A_{\overline{\infty}}$ computed in section 6:

$$\begin{pmatrix} 7.9329\ 7000 & -7.5160\ 9000 & -0.9401\ 2000 & 4. \\ 0.0001\ 0460 & -8.0286\ 4604 & -3.5699\ 6326 & 3.4887\ 6000 \\ -0.0000\ 3790 & 0.0001\ 0570 & 5.6688\ 9760 & -4.0237\ 8459 \\ -0.0000\ 3312 & 0.0000\ 2741 & 0.0000\ 3316 & -1.5732\ 2156 \end{pmatrix}.$$

One sweep gives the following improved A_∞ and Λ_∞:

$$\begin{pmatrix} 7.9329\ 0475 & -7.5160\ 8890 & -0.9401\ 0388 & 4. \\ 0.0000\ 0007 & -8.0285\ 7837 & -3.5699\ 4154 & 3.4887\ 2740 \\ 0.0000\ 0001 & -0.0000\ 0003 & 5.6688\ 6438 & -4.0237\ 3544 \\ 0.0000\ 0004 & 0.0000\ 0002 & 0.0000\ 0002 & -1.5731\ 9076 \end{pmatrix}$$

$$\begin{pmatrix} 1. & 0 & 0 & 0 \\ 0.3778\ 1815 & 1. & 0 & 0 \\ 1.3866\ 2121 & -1.0097\ 5198 & 1. & 0 \\ 0.3488\ 0574 & -1.3717\ 0824 & -0.9850\ 2597 & 1 \end{pmatrix}.$$

Continuing in this way, we may attain any desired accuracy. The same treatment is possible for matrices with conjugate complex latent roots without leaving the domain of real numbers.

Estimates. Instead of improving the result of (35) as indicated by the numerical example above, one may prefer to establish bounds for the latent roots of A. Such bounds may be obtained by Laplacian development of det $(a_{ij}-\lambda\delta_{ij})$, where a_{ij} are the elements of $A_{\overline{\infty}}$. If s is the sum of the absolute values of all subdiagonal elements, and if for the λ's in question all column vectors of $A_{\overline{\infty}}-\lambda E$ are smaller (in length) than a constant M, then clearly

$$|\det(A_{\overline{\infty}}-\lambda E)-\prod_{i=1}^{n}(a_{ii}-\lambda)|<sM^{n-1}. \tag{36}$$

This formula, however, will in general lead to a very poor estimate for the λ's, but it can be improved as follows:

If we denote the length of the kth column vector of $(A_{\overline{\infty}}-\lambda E)$ by $M_k(\lambda)$, and the sum of the absolute values of the subdiagonal elements of the kth column by s_k, then the right side of (36) may be replaced by the better value

$$M_1(\lambda)M_2(\lambda)\ \ldots\ M_n(\lambda)\cdot\sum_{1}^{n-1}\frac{s_j}{M_j(\lambda)}. \tag{37}$$

9. Connections With "Deflation"

It is well known that if a dominant latent root λ_1 of A and the corresponding latent vectors x_1 of A and y_1 of A^T have been found, the latent root λ_1 can be eliminated from A by a procedure called "deflation" (see Bodewig [1], especially first part, p. 169 onwards):

$$A-\lambda_1\frac{(x_1y_1^{\mathrm{T}})}{(x_1^{\mathrm{T}}y_1)}\Longrightarrow A_1. \tag{38}$$

The resulting A_1 has the latent roots $0,\lambda_2,\lambda_3,\ \ldots\ ,\lambda_n$ and the same latent vectors as A. A slightly different procedure allows the transformation of A into a matrix with only $n-1$ rows and columns and the latent roots $\lambda_2,\lambda_3,\ \ldots\ ,\lambda_n$.

Both methods, however, have the disadvantage that by repeated application (in order to compute *all* latent roots) the truncation errors [15] may build up in a dangerous way. For this reason it may be worth mentioning that the methods of section 6 suggest a procedure only slightly different from deflation but not so much suffering from roundoff errors. This method, which is due to G. Blanch,[16] may be defined as follows:

Let $x=(1,x_2,x_3,\ \ldots\ ,x_n)^T$ be an approximation to a latent vector of A, which may have been found by iteration. Then the transformation

$$A_2=L_1^{-1}AL_1,$$

where

$$L_1=\begin{pmatrix} 1 & & & & & \\ x_2 & 1 & & \mathbf{O} & & \\ x_3 & 0 & 1 & & & \\ \cdot & \cdot & & & \cdot & \\ \cdot & \cdot & \mathbf{O} & & & \cdot \\ \cdot & \cdot & & & & & \cdot \\ x_n & 0 & & & & & 1 \end{pmatrix} \tag{39}$$

will practically eliminate the subdiagonal elements of the first column; i. e., we obtain

---

[15] By this we mean errors caused by using in (38) vectors x_1 and y_1, which are not exactly latent vectors.

[16] Unpublished, cited by Feller and Forsythe [13].

$$A_2=\left[\begin{array}{c|cccc|}
* & * & * & & * \\
\hline
\epsilon & * & * & & * \\
\epsilon & * & * & & * \\
\vdots & \vdots & \vdots & & \vdots \\
\epsilon & * & * & & * \\
\hline
\end{array}\right], \tag{40}$$

where * denotes normal, ϵ small elements. After that we determine the dominant latent vector y of the $n-1-$row submatrix which is framed in (40) and normalize it again so that $y=(1,y_3,y_4, \ldots ,y_n)^T$. Then the whole matrix A_2 is transformed with

$$L_2=\begin{bmatrix}
1 & & & & & \\
0 & 1 & & & \mathbf{O} & \\
0 & y_3 & 1 & & & \\
0 & y_4 & 0 & \ddots & & \\
\vdots & \vdots & \vdots & & \ddots & \\
\vdots & \vdots & \vdots & \mathbf{O} & & \ddots \\
0 & y_n & 0 & & & 1
\end{bmatrix},$$

yielding $A_3=L_2^{-1}A_2L_2$, and so on.

By continuing in this way we obtain after $n-1$ steps a matrix A_n, which would be triangular, if the latent vectors x,y used in the transformation matrices $L_1,L_2, \ldots , L_{n-1}$ had been exact. As this practically is never the case, A_n will not be exactly triangular, but can be corrected by the methods of section 6 and (if needed) of section 8.

Trouble will occur, however, as soon as A has complex conjugate roots or rootpairs of otherwise equal or nearly equal absolute value. In the course of the columnwise reduction to triangular form, such a rootpair will become dominant at a certain stage and then the power method will not converge. In analogy to the methods of section 6, the following procedure is suggested in such a case:

An extension of the method of G. Blanch. Let A be a matrix with p dominant latent roots of equal or nearly equal absolute value, and let $x_k,x_{k-1},x_{k-2}, \ldots ,x_{k-p+1}$ be p succeeding iteration vectors (in the sense of Von Mises-Geiringer). If we write their components as columns of an $n\times p$-matrix, and if the Gauss-Banachiewicz elimination procedure (see footnote 6) is applied to this $n\times p$-matrix, an "incomplete lower triangle" L' and an upper triangle R' are obtained:[17]

$$L'=\begin{bmatrix}
1 & & & \\
x_2 & 1 & \mathbf{O} & \\
x_3 & y_3 & 1 & \\
 & & \ddots & \\
\vdots & \vdots & \vdots & 1 \\
\vdots & \vdots & \vdots & \vdots \\
x_n & y_n & \cdots & w_n
\end{bmatrix}, \qquad
R'=\begin{bmatrix}
* & * & * & \cdots & * \\
0 & * & * & \cdots & * \\
0 & 0 & * & \cdots & * \\
\vdots & & & \ddots & \vdots \\
\vdots & \mathbf{O} & & & * \\
\vdots & & & & \vdots \\
0 & 0 & 0 & \cdots & 0
\end{bmatrix}.$$

---

[17] For large k, such a triangular decomposition will fail as soon as the p largest latent roots differ too much in absolute value.

Then we extend the matrix L' to a square matrix by adding $n-p$ unit column-vectors; this gives a transformation matrix

$$L_k=\left(\begin{array}{ccccccccc}
1 &&&&&&&& \\
x_2 & 1 &&&&&&& \\
x_3 & y_3 & 1 &&&& \mathbf{O} && \\
&&& \ddots &&&&& \\
\cdot & \cdot & \cdot && 1 &&&& \\
\cdot & \cdot & \cdot && \cdot & 1 &&& \\
\cdot & \cdot & \cdot && \cdot & 0 & \ddots && \\
&&&&&&& 1 & \\
x_n & y_n & \cdots\cdots & & w_n & 0 & \cdots & 0 & 1
\end{array}\right), \tag{41}$$

which has for large k the property

$$L_k^{-1}AL_k=\left(\begin{array}{c|c}
\begin{array}{ccc}*&*&*\\*&*&*\\*&*&*\end{array} & \left.\begin{array}{cc}*&*\\*&*\\*&*\end{array}\right\}p \\ \hline
\underbrace{\begin{array}{ccc}\epsilon&\epsilon&\epsilon\\\epsilon&\epsilon&\epsilon\end{array}}_{p} & \underbrace{\left.\begin{array}{cc}*&*\\*&*\end{array}\right\}n-p}_{n-p}
\end{array}\right), \tag{42}$$

where again a star denotes a normal, ϵ a small element. After that, the submatrix in the right lower corner of (42) may be treated in the same way. A numerical example may illustrate the effect of the transformation (42). Let A be Bodewig's matrix

$$A=\begin{pmatrix} 2 & 1 & 3 & 4 \\ 1 & -3 & 1 & 5 \\ 3 & 1 & 6 & -2 \\ 4 & 5 & -2 & -1 \end{pmatrix}.$$

Beginning with $v_0=(0|0|1|1)$, we obtain by iteration:

v_0	v_1	v_2	. . .	v_7	v_8	v_9
0	7	20	. . .	1 581 057	4 608 988	99 985 809
0	6	−22		1 655 272	−6 683 198	105 805 544
1	4	57		1 104 706	14 787 937	69 327 778
1	−3	53		−880 629	13 271 805	−57 827 717

There is no indication of convergence, therefore (42) is applied. Triangular decomposition of the 3×4 matrix with columns $v_9v_8v_7$ gives

99 985 809	4 608 988	1 581 057
−1.058 205 610	−11 560 454.98	− 17 811.387
−0.693 376 177	1.002 743 797	− 9 421.516
0.578 359 245	1.378 618 389	0.980 184 831

Here the comparatively small element -9421 indicates that there are only two dominant latent roots. Therefore we take $p=2$, and obtain the transformation matrix

$$L=\begin{pmatrix} 1. & 0 & 0 & 0 \\ 1.058206 & 1. & 0 & 0 \\ 0.693376 & -1.002744 & 1 & 0 \\ -0.578359 & -1.378618 & 0 & 1 \end{pmatrix}$$

and

$$L^{-1}AL=\left(\begin{array}{rr|rr} 2.824898 & \multicolumn{1}{r}{-7.522704} & 3. & 4. \\ -7.362361 & \multicolumn{1}{r}{-2.935264} & -2.174618 & 0.767176 \\ \cline{1-2} 0.033901 & 0.013576 & 1.739287 & -4.004223 \\ -0.033441 & -0.013325 & -3.262891 & 2.371079 \end{array}\right).$$

Completion of the transformation into triangular form may be done with the methods of section 6 and (if needed) section 8.

10. Determination of Latent Roots of "Striped" Matrices

As every single LR-transformation step requires the triangular decomposition of an $n\times n$ matrix, the computation labor for the determination of latent roots by the LR-transformation seems excessive, and the reader may have the impression that the numerical examples of this report could have been solved more easily by other methods (which is certainly true for four-row matrices).

However, there are classes of matrices for which the LR-transformation is far superior to any other method. These are the striped matrices that occur frequently in numerical applications.

Definition: A matrix $A=(a_{ij})$ is called a striped matrix, if there exists a number m so that

$$a_{ij}=0 \quad \text{for} \quad |i-j|>m. \tag{43}$$

A typical example is the difference equation for the vibrating beam; the corresponding matrix is of the form (43) with $m=2$. For a clamped homogeneous beam we have roughly

$$A=\begin{pmatrix} 6 & -4 & 1 & & & & & \\ -4 & 6 & -4 & 1 & & & \mathbf{0} & \\ 1 & -4 & 6 & -4 & 1 & & & \\ & 1 & -4 & 6 & \cdot & \cdot & & \\ & & \cdot & \cdot & \cdot & \cdot & \cdot & \\ & & & \cdot & \cdot & \cdot & \cdot & \cdot \\ & & & & \cdot & \cdot & -4 & 1 \\ & \mathbf{0} & & & 1 & -4 & 6 & -4 \\ & & & & & 1 & -4 & 6 \end{pmatrix}. \tag{44}$$

For matrices of the type (43), the triangular matrices L and R have the same property, namely,

$$\begin{aligned} l_{ij}&\neq 0 \quad \text{only for} \quad i-m\leq j\leq i \\ r_{ij}&\neq 0 \quad \text{only for} \quad i\leq j\leq i+m, \end{aligned} \tag{45}$$

and therefore (43) will hold also for $A_2=RL$ and for all succeeding A_k, i. e.,

THEOREM 7. *Property* (43) *is maintained by the LR-transformation.*

This is of primary importance, because the computational labor for the triangular decomposition is much smaller for matrices of type (43) than for full matrices, provided m is not too large.

If the matrix A is symmetric and positive definite (which is very often the case in applications of that type), and if we look only for latent roots and not for the vectors, a further saving is possible by using a symmetric decomposition of Choleski in place of the nonsymmetric Gauss-Banachiewicz decomposition (1):

$$\begin{aligned} A_k &\Longrightarrow L_k L_k^T \\ L_k^T L_k &\Longrightarrow A_{k+1} \end{aligned} \tag{46}$$

As this procedure preserves the symmetry, the storage capacity needed is again reduced at the expense of computing n square roots. As an example: For $n=100$, $m=2$, one single LR step will take about 2½ min. on the electronic computer ERMETH [14], and 297 storage positions will be needed for the numbers. On the other hand, the Jacobi method [2,3] will destroy property (43) and therefore 5,050 storage positions would be needed for the elements of the matrices A_k.

The convergence of method (46) can be proved by the same type of argument as used in section 3. There is, however, a simpler proof based on

THEOREM 8. *Let $D_{k,p}$ be the pth principal minor of the matrix A_k, which has been obtained from a positive definite symmetric matrix A_1 by procedure* (46). *Then*

$$D_{1,p} \leq D_{2,p} \leq D_{3,p} \leq \ldots \leq \lambda_1 \lambda_2 \ldots \lambda_p \tag{47}$$

for any $p=1,2,\ldots,n$.

PROOF. By virtue of the decomposition formula $A_k \Longrightarrow L_k L_k^T$, the pth principal minor of L_k is the square root of the corresponding minor of A_k. On the other hand, the pth principal minor of $A_{k+1}=L_k^T L_k$ is the sum of the squares of certain subdeterminants of L_k, one of them being its pth principal minor $\sqrt{D_{k,p}}$, q. e. d.

From (47) we infer that $\lim\limits_{k\to\infty} D_{k,p}$ must exist for all p, which is possible only if L_k converges to a diagonal matrix. Therefore

THEOREM 9. *If a symmetric and positive definite matrix $A=A_1$ is treated by the LR-transformation with symmetric decomposition* (46), *all A_k are symmetric and $\lim\limits_{k\to\infty} A_k$ is a diagonal matrix.*

As a numerical example we show the treatment of the matrix (44) with $n=15$ by this method. After five LR steps we obtain

$$A_6=\left[\begin{array}{cccccccc}
\underline{12.71272} & -2.23291 & 0.17605 & & & & & \\
-2.23291 & \underline{10.23820} & -3.00522 & & & & & \\
0.17605 & -3.00522 & \ddots & & & & & \\
 & & & \ddots & & & & \\
 & & & & \underline{1.966537} & -0.349106 & 0.004933 & 0 \\
 & & & & -0.349106 & \underline{0.288728} & -0.021173 & 0.000093 \\
 & & & & 0.004933 & -0.021173 & \underline{0.047707} & -0.000642 \\
 & & & & 0 & 0.000093 & -0.000642 & \underline{0.006024}
\end{array}\right]$$

$$\lambda:\ (\sim 15.70) \qquad (\sim 14.85) \qquad\qquad (0.445216) \quad (0.169168) \quad (0.045124) \quad (0.006014)$$

(Here the diagonal elements are underlined.) Comparing with the "exact" values of the four smallest latent roots (below the columns of A_6), we see at once that the lower diagonal elements converge faster to the corresponding latent roots than the other ones, i. e., for matrices of this type the smallest latent roots "appear" first. This phenomenon is of enormous practical importance, because such matrices originate frequently from eigenvalue problems for differential equations where in general only the smallest latent roots are interesting.

Therefore, we may apply the LR-transformation for striped matrices in many cases without using the convergence-improving methods of section 6 which would destroy property (43). There is, however, a very simple method to speed up the convergence without destroying property (43). It is based on the observation that the element a_{nn} of the matrix A_k converges for $k\to\infty$ to λ_n, roughly as $(\lambda_n/\lambda_{n-1})^k$ converges to zero. This means that if we apply the LR-transformation to the matrix $A-xE$ where $0<x<\lambda_n$, a_{nn} will converge faster than before, but now to the value $\lambda_n - x$.

In order to obtain optimal convergence, x must be chosen as near to λ_n as possible. On the other hand, $\lambda_n - x$ should not be negative, because then the Cholesky procedure (46) would lead to imaginary numbers, which is undesirable. One method is to estimate a lower bound for the latent roots with the method of S. Gershgorin [16] and to use this lower bound as x. But numerical experiments with the matrix (44) have shown that it requires many LR steps until Gershgorin's formula gives a positive lower bound. Indeed, for the matrix (44) the lower bound resulting from Gershgorin's formula is -4, and even for A_6 it is still near to -2.5. But we can find a much better lower bound by the following method.

From the triangular decomposition of $A-\lambda E$ we obtain immediately one value of the function $D(\lambda)$, the zeros of which are the latent roots of A. If we carry out the decomposition for two different values λ and μ, we obtain two values $D(\lambda)$ and $D(\mu)$ of this function from which we can construct a secant and its intersection x with the λ-axis. As long as μ,λ are smaller than λ_n (for instance, $\lambda=0$, $\mu=-\epsilon$), this secant does not intersect the curve $\{\lambda,D(\lambda)\}$ a third time, and therefore x is a lower bound for λ_n:

$$x=\frac{\lambda D(\mu)-\mu D(\lambda)}{D(\mu)-D(\lambda)}. \tag{48}$$

To summarize, we obtain the following procedure:
Compute[18] for $k=0,1,2,\ldots,$

$$\frac{x_{k-2}D(x_{k-1})-x_{k-1}D(x_{k-2})}{D(x_{k-1})-D(x_{k-2})}\Longrightarrow x_k \tag{49}$$

$$\left.\begin{array}{l} A_k-x_kE \text{ decomposed}\Longrightarrow L_kL_k^T \\[1ex] \text{Product of the squares of} \\ \qquad \text{the diagonal elements of } L_k\Longrightarrow D(x_k) \\[1ex] x_kE+L_k^TL_k\Longrightarrow A_{k+1}. \end{array}\right\} \tag{50}$$

As well as the LR-transformation defined by formulas (46) this new procedure gives a sequence of symmetric matrices that are similar to each other. But it has the advantage that the last diagonal element converges much faster to λ_n and that very soon the outdiagonal elements of the last row and column become negligible. When this point is reached, the last diagonal element is (practically) a latent root. So we leave out the last row and column and proceed in the same way with the remaining $n-1$-row matrix. In this way we get the latent roots, beginning with the smallest, one after the other with increased speed.

11. A Continuous Analog to the LR-Transformation

In the foregoing section we have seen that the LR-transformation can be influenced by a shift of origin in the plane of the latent roots. In this section we choose a shift of origin far to the left, i.e., we choose $x=-M$, where M is a large positive number. This gives the following procedure:

$$\begin{array}{ll} A_k+ME \text{ decomposed} & \Longrightarrow L_kR_k, \\[1ex] R_kL_k-ME & \Longrightarrow A_{k+1}, \end{array} \tag{51}$$

where the decomposition should be of Gauss-Banachiewicz type (see footnote 6).

This, of course, slows down the convergence, because the difference between A_{k+1} and A_k tends to zero for $M\rightarrow\infty$. Indeed, if we decompose A_k+ME for large M with method (1), then

$$\begin{array}{l} L_k=E+\dfrac{X}{M}+\dfrac{P}{M^2}+\text{higher terms} \\[2ex] R_k=ME+Y+\dfrac{Q}{M}+\text{higher terms} \end{array} \tag{52}$$

---

[18] For the first two steps ($k=0,k=1$), x_k cannot be determined by (49). Instead the values $x_0=-\epsilon$ and $x_1=0$ may be used.

where X and Y are left and right triangular matrices with all diagonal elements of X equal to zero and $X+Y=A_k$. P and Q are defined as left and right triangular matrices with $P+Q=-XY$, all diagonal elements of P being zero. Therefore

$$A_{K+1}-A_k=R_kL_k-L_kR_k$$

$$=(ME+Y+\frac{Q}{M}+\ .\ .\ .)(E+\frac{X}{M}+\frac{P}{M^2}+\ .\ .\ .)-(E+\frac{X}{M}+\frac{P}{M^2}+\ .\ .\ .)(ME+Y+\frac{Q}{M}+\ .\ .\ .)$$

$$=\frac{1}{M}(YX-XY)+\frac{1}{M^2}(QX+PY-XQ-YP)+\text{higher terms.}$$

Going to the limit $M\to\infty$, $(k/M)\to t$ and denoting A_k by $A(t)$, we obtain the following differential equation for $A(t)$:

$$\frac{dA}{dt}=YX-XY, \quad \text{with} \quad X+Y=A(t), \tag{53}$$

where X and Y are left and right triangular matrices, the diagonal elements of X being zero.

If this differential equation is integrated with the initial condition $A(0)=A$, a continuous sequence of similar matrices is obtained for which—as a consequence of theorem 3—the following theorem holds:

THEOREM 10. *If the latent roots* λ_i *of* A *fulfill the conditions*

$$\mathrm{Re}(\lambda_1)>\mathrm{Re}(\lambda_2)>\ .\ .\ .\ >\mathrm{Re}(\lambda_n), \tag{54}$$

and if (14b) *holds for the matrices* U *and* V *defined in* (11), *then* $\lim_{t\to\infty} A(t)$ *exists and is an upper triangular matrix.*

It may be noted that the corresponding matrices $\Lambda(t)$ and $\mathrm{P}(t)$, which transform $A(0)$ into $A(t)$,

$$A(t)=\Lambda^{-1}(t)A(0)\Lambda(t)=\mathrm{P}(t)A(0)\mathrm{P}^{-1}(t), \tag{55}$$

are solutions of the following matrix-differential equations:

$$\frac{d\Lambda}{dt}=\Lambda(t)X(t), \qquad \Lambda(0)=E$$

$$\frac{d\mathrm{P}}{dt}=Y(t)\mathrm{P}(t), \qquad \mathrm{P}(0)=E. \tag{56}$$

From this it follows that

$$\frac{d(\Lambda)\mathrm{P}}{dt}=\Lambda X(t)\mathrm{P}+\Lambda Y(t)\mathrm{P}=\Lambda A(t)\mathrm{P}=A(0)\Lambda\mathrm{P},$$

or

THEOREM 11. *The matrices* $\Lambda(t)$ *and* $\mathrm{P}(t)$ *defined in* (55) *can be obtained by triangular decomposition of* $e^{tA(0)}$:

$$e^{tA(0)}\Longrightarrow\Lambda(t)\mathrm{P}(t). \tag{57}$$

For hermitian matrices (they need not be positive definite), the severe restrictions of theorem 10 can be weakened considerably: *If* $A(0)=A$ *is hermitian, then* $\lim A(t)$ *will always exist.*

For symmetric matrices it is possible to use a symmetric additive decomposition $A(t)=X+X^T$ in place of the additive decomposition $A(t)=X+Y$ used in (53). If this is done, dA/dt is symmetric too, and therefore $A(t)$ *will be symmetric for all* t and—as a consequence of theorem 9—$\lim A(t)$ *will exist and be diagonal.*

12. A Graeffe-Like Modification of the LR-Transformation[19]

From formula (9), Dr. F. L. Bauer and the present author have been led to the idea that the decomposition of A^{2k} probably might be computed from the decomposition of A^k. Indeed if $A^k=\Lambda_k\mathrm{P}_k$, then $A^{2k}=\Lambda_k\mathrm{P}_k\Lambda_k\mathrm{P}_k$. Thus, if $\mathrm{P}_k\Lambda_k$ is decomposed again into a left and right triangular matrix (see footnote 6), $\mathrm{P}_k\Lambda_k=\Lambda'_k\mathrm{P}'_k$, then obviously

$$A^{2k}=\Lambda_k\Lambda'_k\mathrm{P}'_k\mathrm{P}_k \quad \text{or} \quad \Lambda_{2k}=\Lambda_k\Lambda'_k, \quad \mathrm{P}_{2k}=\mathrm{P}'_k\mathrm{P}_k. \tag{58}$$

This formula enables us to skip the computation of $A_k,\Lambda_k,\mathrm{P}_k$ for $k\neq 2^n$. The procedure (58) suffers, however, from the very large and very small numbers involved in the computations. Indeed, as $\Lambda_k\mathrm{P}_k$ is the decomposition of A^k for $k=2^n$, the diagonal elements of P_k are approximately the 2^nth powers of the latent roots.

In order to avoid the large numbers, we define matrices Σ_k and D_k, where Σ_k is a right triangular matrix with diagonal elements 1 and D_k is diagonal, so that $\mathrm{P}_k=D_k\Sigma_k$. This leads to $D_k\Sigma_k\Lambda_k$ in place of $\mathrm{P}_k\Lambda_k$, and if we decompose[20]

$$\Sigma_k\Lambda_k\Longrightarrow\Lambda^*_kD^*_k\Sigma^*_k,$$

we have

$$\Lambda'_k\mathrm{P}'_k=\mathrm{P}_k\Lambda_k=D_k\Sigma_k\Lambda_k=D_k\Lambda^*_kD^*_k\Sigma^*_k.$$

But as Λ'_k is a left triangular matrix with diagonal elements 1, we find that $\Lambda'_k=D_k\Lambda^*_kD_k^{-1}$ and $\mathrm{P}'_k=D_kD^*_k\Sigma^*_k$, which gives immediately

$$\left.\begin{array}{ll}
\Lambda_{2k}=\Lambda_k\Lambda'_k=\Lambda_kD_k\Lambda^*_kD_k^{-1} & \\
\mathrm{P}_{2k}=\mathrm{P}'_k\mathrm{P}_k=D_kD^*_k\Sigma^*_kD_k\Sigma_k & (=D_{2k}\Sigma_{2k}), \\
\text{and therefore (see footnote 20)} & \\
\Sigma_{2k}=D_k^{-1}\Sigma^*_kD_k\Sigma_k & \\
D_{2k}=D_k^2\cdot D^*_k. &
\end{array}\right\} \tag{59}$$

We have still the large numbers in the diagonal matrices D_k, but we can eliminate even these by introducing the matrix

$$\left.\begin{array}{l}
H(D)=D\begin{pmatrix} 1 & & & \\ 1 & 1 & & \mathbf{0} \\ \vdots & \vdots & \ddots & \\ 1 & 1 & \cdots & 1 \end{pmatrix}D^{-1} \\
\text{as a substitute for the diagonal matrix } D. \text{ Then} \\
D^{-1}\Sigma D=\{H^T(D)\cdot\Sigma\} \\
D\Lambda D^{-1}=\{H(D)\cdot\Lambda\},
\end{array}\right\} \tag{60}$$

where $\{X\cdot Y\}$ denotes the "elementwise product" of two matrices.

The combination of (59) and (60) leads to the final formulas.

---

[19] See also footnote 1 and reference [8]. In the meantime, F. L. Bauer has developed some important generalizations of the LR-transformations, the most interesting one being the "Bi-Iteration" (see [18], especially sec. 4).

[20] In the sequel, Σ and Λ shall always denote right and left triangular matrices *with diagonal elements* 1, and D shall be diagonal.

Begin with the triangular decomposition of A (see footnote 20):

$$\left.\begin{array}{r} A \Longrightarrow \Lambda_1 D_1 \Sigma_1 \\ H(D_1) \Longrightarrow H_1. \end{array}\right\} \tag{61}$$

Then compute for $k=1,2,4,8, \ldots, 2^n$:

$$\left.\begin{array}{ll} \text{(a)} & \Sigma_k \Lambda_k \Longrightarrow \Theta_k \\ \text{(b)} & \text{decompose } \Theta_k \Longrightarrow \Lambda_k^* D_k^* \Sigma_k^* \\ \text{(c)} & H(D_k^*) \Longrightarrow H_k^* \\ \text{(d)} & \Lambda_k \{ H_k \cdot \Lambda_k^* \} \Longrightarrow \Lambda_{2k} \\ \text{(e)} & \{ H_k^T \cdot \Sigma_k^* \} \Sigma_k \Longrightarrow \Sigma_{2k} \\ \text{(f)} & \{ H_k^* \cdot H_k \cdot H_k \} \Longrightarrow H_{2k}. \end{array}\right\} \tag{62}$$

Stop, when $\Sigma_k \approx \Sigma_{2k}$ and $\Lambda_k \approx \Lambda_{2k}$, and compute

$$\Lambda_k^{-1} A \Lambda_k \Longrightarrow A_k = A_\infty \tag{63}$$

In this way, the large numbers are entirely eliminated and no 2^nth root has to be computed to obtain the latent roots. This is a decided advantage over Graeffe's method, especially if there are conjugate complex roots.

It should be noted that this method to accelerate the convergence does not preserve property (43), therefore it should not be applied to striped matrices with large n and small m.

Numerical example. We take the matrix

$$A = \begin{pmatrix} 20 & -7 & -7 & 2 \\ -7 & 12 & 2 & -5 \\ -7 & 2 & 12 & -5 \\ 2 & -5 & -5 & 6 \end{pmatrix},$$

which is critical insofar as it has a very close pair of latent roots, 10 and 10.023 . . . . We obtain

$$\operatorname{dec} A = \begin{pmatrix} 20. & -7. & -7. & 2. \\ 0.35 & 9.55 & -0.45 & -4.3 \\ 0.35 & 0.047120 & 9.528796 & -4.502618 \\ -0.1 & 0.450262 & 0.472527 & 1.736265 \end{pmatrix}$$

$$\Lambda_1 = \begin{pmatrix} 1. & 0 & 0 & 0 \\ -0.35 & 1. & 0 & 0 \\ -0.35 & -0.047120 & 1. & 0 \\ 0.1 & -0.450262 & -0.472527 & 1 \end{pmatrix} = \Sigma_1^T$$

$$D_1 = \begin{pmatrix} 20 & 0 & 0 & 0 \\ 0 & 9.55 & 0 & 0 \\ 0 & 0 & 9.528796 & 0 \\ 0 & 0 & 0 & 1.736265 \end{pmatrix}$$

$$H_1 = \begin{pmatrix} 1. & 0 & 0 & 0 \\ 0.4775 & 1. & 0 & 0 \\ 0.476440 & 0.997780 & 1. & 0 \\ 0.086813 & 0.181808 & 0.182212 & 1 \end{pmatrix}.$$

First step:

$$\Theta_1=\begin{pmatrix} 1.255 & -0.378534 & -0.397253 & 0.1 \\ -0.378534 & 1.204956 & 0.165641 & -0.450262 \\ -0.397253 & 0.165641 & 1.223282 & -0.472527 \\ 0.1 & -0.450262 & -0.472527 & 1. \end{pmatrix}$$

$$\text{dec }\Theta_1=\begin{pmatrix} 1.255 & -0.378534 & -0.397253 & 0.1 \\ 0.301621 & 1.090782 & 0.045821 & -0.420100 \\ 0.316536 & -0.042008 & 1.095612 & -0.423226 \\ -0.079681 & 0.385137 & 0.386292 & 0.666747 \end{pmatrix}$$

$$\Lambda_1^*=\begin{pmatrix} 1. & 0 & 0 & 0 \\ -0.301621 & 1. & 0 & 0 \\ -0.316536 & 0.042008 & 1. & 0 \\ 0.079681 & -0.385137 & -0.386292 & 1 \end{pmatrix}=\Sigma_1^{*T}$$

$$\{H_1.\Lambda_1^*\}=\begin{pmatrix} 1. & 0 & 0 & 0 \\ -0.144024 & 1. & 0 & 0 \\ -0.150810 & 0.041915 & 1. & 0 \\ 0.006917 & -0.070021 & -0.070387 & 1 \end{pmatrix}$$

$$H_1^*=\begin{pmatrix} 1. & 0 & 0 & 0 \\ 0.869149 & 1. & 0 & 0 \\ 0.872998 & 1.004428 & 1. & 0 \\ 0.531273 & 0.611256 & 0.608561 & 1 \end{pmatrix}$$

$$\Lambda_2=\begin{pmatrix} 1. & 0 & 0 & 0 \\ -0.494024 & 1. & 0 & 0 \\ -0.494024 & -0.005205 & 1. & 0 \\ 0.243027 & -0.540089 & -0.542914 & 1 \end{pmatrix}=\Sigma_2^T$$

$$H_2=\begin{pmatrix} 1. & 0 & 0 & 0 \\ 0.198171 & 1. & 0 & 0 \\ 0.198166 & 0.999973 & 1. & 0 \\ 0.004003 & 0.020204 & 0.020205 & 1 \end{pmatrix}.$$

This completes the first step. Further calculation yields:

$$\Lambda_4=\begin{pmatrix} 1. & 0 & 0 & 0 \\ -0.573784 & 1. & 0 & 0 \\ -0.573784 & 0.029448 & 1. & 0 \\ 0.330262 & -0.567486 & -0.551252 & 1 \end{pmatrix}=\Sigma_4^T$$

$$\Lambda_8=\begin{pmatrix} 1. & 0 & 0 & 0 \\ -0.585417 & 1. & 0 & 0 \\ -0.585417 & 0.039171 & 1. & 0 \\ 0.343089 & -0.572947 & -0.551349 & 1 \end{pmatrix}$$

and so on. Finally,

$$\Lambda_{8192}=\begin{pmatrix} 1. & 0 & 0 & 0 \\ -0.585590 & 1. & 0 & 0 \\ -0.585590 & 0.999808 & 1. & 0 \\ 0.343280 & -1.102594 & -0.551349 & 1 \end{pmatrix}$$

$$A_{8192}=\begin{pmatrix} 28.884820 & -16.203844 & -8.102698 & 2. \\ 0.000002 & 10.023777 & 0.011886 & -3.828820 \\ 0 & 0.000005 & 10.000002 & -0.000735 \\ 0.000001 & 0.000011 & -0.000030 & 1.091397 \end{pmatrix}.$$

This result shows that the method has a remarkable numerical stability.

13. Appendix. Numerical Experiments With Striped Matrices[21]

In order to obtain information about the speed and stability of the LR-transformation, the latent roots of some striped matrices with large n were computed with the electronic computer ERMETH [14] of the Swiss Federal Institute of Technology.

As a first example, matrix (44) with $n=50$ was treated. The latent roots were computed with the routine described by formulas (49), (50) of section 10 and with fixed decimal point.[22]

	Results obtained with 12 digits after decimal point	Results obtained with 10 digits after decimal point	Latent roots of the continuous problem (65)
$\lambda_{50}=$	0.000068 487899	0.000068 4877	0.000068 4615
$\lambda_{49}=$	0.000519 731902	0.000519 7317	0.000520 2047
$\lambda_{48}=$	0.001992 423727	0.001992 4234	0.001999 1329
$\lambda_{47}=$	0.005424 127619	0.005424 1268	0.005463 0603

(64)

These numbers indicate the magnitude of the errors that must be expected in problems of that kind. Furthermore, one can compare them with the exact roots of the continuous problem from which matrix (44) was derived:

$$y^{iv}=\lambda y, \qquad y(-0{,}5)=y'(-0{,}5)=y(51{,}5)=y'(51{,}5)=0. \tag{65}$$

We see that the truncation errors are far greater than the round-off errors, so that an optimal result already can be obtained in fewer LR-steps (here about 10 steps for the first 4 latent roots).

A second experiment was carried out with the 11-row matrix

$$\left.\begin{array}{l}
A=\begin{pmatrix}
5 & 2 & 1 & 1 & \cdot & & & \\
2 & 6 & 3 & 1 & \cdot & & \mathbf{0} & \\
1 & 3 & 6 & \cdot & & \cdot & \cdot & \\
1 & \cdot & & \cdot & \cdot & & & 1 \\
 & \cdot & \cdot & & & 6 & 3 & 1 \\
 & \mathbf{0} & \cdot & & 1 & 3 & 6 & 2 \\
 & & & \cdot & 1 & 1 & 2 & 5
\end{pmatrix} \\[2ex]
\text{or} \qquad a_{ik}=\begin{cases} 6 & \text{for } i=k \\ 3 & \text{for } |i-k|=1 \\ 1 & \text{for } |i-k|=2 \\ 1 & \text{for } |i-k|=3 \\ 0 & \text{for } |i-k|>3 \end{cases} \\[2ex]
\text{with the exceptions} \quad a_{11}=a_{11,11}=5, \\
\qquad\qquad\qquad\qquad\quad a_{12}=a_{21}=a_{10,11}=a_{11,10}=2.
\end{array}\right\} \tag{66}$$

---

[21] Added in proof.

[22] The ERMETH has built-in floating decimal arithmetic (with 11 digits in the mantissa and 3 for the exponent), but can compute as well with fixed decimal point and 14 decimal digits.

The latent roots of this matrix can be given explicitly. The following table (67) compares the exact ones with these obtained with the LR-transformation (using formulas (49,50) of section 10 and with a total of 64 LR-steps):

"Exact" λ's	Computed λ's
$\lambda_1=14.941\ 819\ 328$	14.941 819 341
$\lambda_2=12.196\ 152\ 423$	12.196 152 446
$\lambda_3=8.828\ 427\ 1247$	8.828 427 1356
$\lambda_4=6.$	6.000 000 0074
$\lambda_5=4.406\ 649\ 9007$	4.406 649 9040
$\lambda_6=4.129\ 248\ 4842$	4.129 248 4880
$\lambda_7=4.$	4.000 000 0036
$\lambda_8=4.$	4.000 000 0030
$\lambda_9=3.171\ 572\ 8753$	3.171 572 8780
$\lambda_{10}=1.803\ 847\ 5773$	1.803 847 5784
$\lambda_{11}=0.522\ 282\ 2875$	0.522 282 2876

(67)

This table seems to indicate that the LR-transformation is sufficiently safe even when there are double roots and close root pairs.

For comparison the same matrix has been treated with Jacobi's method [2,3], and it turned out that in the case where $n=11$, $m=3$, the two methods not only give the same accuracy but also take the same time for the determination of all latent roots.

However, the situation will be quite different for larger n, expecially if only a few latent roots at the lower end of the spectrum are required, because for a striped matrix with general m and n the computation of each latent root (beginning with the smallest) takes about $3m^2n$ seconds whereas the Jacobi method always requires the determination of all latent roots, which takes approximately $3n^3$ seconds on the ERMETH.

As an illustration let us take a problem of organic chemistry where the five lowest latent roots of a number of symmetric striped matrices with $n=20$, $m=3$ had to be computed. With the LR-transformation the computing time was 30 minutes for each of these matrices against 7½ hours (extrapolated) with the Jacobi method.

It may be worthwhile to mention that we have also tried to compute the latent roots of matrix (66) from the characteristic polynomial. But although the *exact* characteristic polynomial had been used and the computation had been carried out with 11 significant figures, it was not possible to obtain more than 3 correct digits of the root $\lambda_6=4.129\ldots$.

The last experiment was carried out with the matrix

$$A=\begin{pmatrix}
14 & 14 & 6 & 1 & & & & \\
14 & 20 & 15 & 6 & 1 & & \mathbf{O} & \\
6 & 15 & \ddots & & & \ddots & & \\
1 & & \ddots & & & & \ddots & 1 \\
 & \ddots & & \ddots & & & 15 & 6 \\
 & \mathbf{O} & & & \ddots & 15 & 20 & 14 \\
 & & & & 1 & 6 & 14 & 14
\end{pmatrix}\quad \text{with } n=89. \tag{68}$$

This matrix is simply the third power of

$$J=\begin{bmatrix} 2 & 1 & & & & \\ 1 & 2 & 1 & & \mathbf{O} & \\ & 1 & 2 & 1 & & \\ & & \cdot & \cdot & \cdot & \\ & \mathbf{O} & & 1 & 2 & 1 \\ & & & & 1 & 2 \end{bmatrix}$$

so that exact values of the latent roots of A can be given:

$$\lambda_K=64\cos^6\left(\frac{\pi K}{2(n+1)}\right) \qquad (K=1, \ldots, n). \tag{69}$$

The computation of these roots begins by adding (or subtracting) some numbers to (or from) the original diagonal values of A. However, as these are greater than 10, and the computation is done with 11 decimal floating arithmetic, the last decimal digit carried was the ninth after the decimal point, so that errors of the magnitude 10^{-9} have to be expected. The following table gives the latent roots of matrix (68) as computed with the LR-transformation as well as their differences against the exact values (69).

Computed Values	Errors	Number of LR-steps required
$\lambda_{89}=0.000\ 000\ 001 \mid 869\ 125\ 1244$	$0.061\cdot 10^{-9}$	5
$\lambda_{88}=0.000\ 000\ 115 \mid 646\ 127\ 11$	$0.009\cdot 10^{-9}$	9
$\lambda_{87}=0.000\ 001\ 315 \mid 576\ 084\ 6$	$0.409\cdot 10^{-9}$	13
$\lambda_{86}=0.000\ 007\ 374 \mid 157\ 551\ 6$	$0.430\cdot 10^{-9}$	17
$\lambda_{85}=0.000\ 028\ 052 \mid 024\ 342$	$0.529\cdot 10^{-9}$	21
$\lambda_{84}=0.000\ 083\ 481 \mid 228\ 485$	$0.281\cdot 10^{-9}$	26
$\lambda_{83}=0.000\ 209\ 675 \mid 213\ 26$	$0.141\cdot 10^{-9}$	30

In view of the very bad condition of this matrix, the results not only are excellent, but they are better than the most optimistic guesses. But still more astonishing is the fact that these results could be obtained from intermediate results with large errors. For instance, after one LR-step the last diagonal element $a_{89,89}$ is 0.000064287 in place of the exact value 0.000061738 . . . . Since the latent root λ_{89} is computed from $a_{89,89}$ simply by subtracting some numbers, it is hard to explain how the large error of $2500\cdot 10^{-9}$ is canceled out, yet the same phenomenon occurs regularly with matrices of that type.

14. References

[1] E. Bodewig, Bericht über die Methoden zur numerischen Lösung algebraischer Eigenwertprobleme, I & II, Atti seminar. mat. e fis. univ. Modena **4,** 133–193 (150); **5,** 3–39 (1951).

[2] R. T. Gregory, Computing eigenvalues and eigenvectors of a symmetric matrix on the ILLIAC, Math. Tables and Other Aids to Computation, **7,** 215–220 (1953).

[3] C. G. J. Jacobi, Ein leichtes Verfahren, die in der Theorie der Säkularstörungen vorkommenden Gleichungen numerisch aufzulosen, J. reine angew. Math. **30,** 51–95 (1846).

[4] G. Kowalewski, Einführung in die Determinantentheorie (Verlag Voit & Comp., Leipzig, 1909).

[5] H. Rutishauser, Der Quotienten-Differenzen-Algorithmus, Z. angew. Math. u. Phys. **5,** 233–251 (1954).

[6] H. Rutishauser, Anwendungen des Quotienten-Differenzen-Algorithmus, Z. angew. Math. u. Phys. **5,** 496–508 (1954).

[7] H. Rutishauser, Une méthode pour la détermination des valeurs propres d'une matrice, Compt. rend. Paris **240,** 34 (1955).

[8] H. Rutishauser and F. L. Bauer, Détermination des vecteurs propres d'une matrice par une méthode iterative avec convergence quadratique, Compt. rend. Paris **240,** 1681 (1955).

[9] H. Rutishauser, Automatische Rechenplanfertigung bei programmgesteuerten Rechenmaschinen, Rept. No. 3, Inst. angew. Math. der Eidgenössische Technische Hochschule (Verlag Birkhäuser, Basel, 1952).

[10] R. Zurmühl, Praktische Mathematik (Springer Verlag, 1953).
[11] E. Bodewig, A practical refutation of the iteration method for the algebraic eigenvalue problem, Math. Tables and Other Aids to Computation **8**, 237–240 (1954).
[12] R. v. Mises and H. Geiringer, Praktische Verfahren zur Gleichungsauflosung, Zusammenfassender Bericht, Z. angew. Math. Meth. **9**, 58–77 and 152–164 (especially p. 154) (1929).
[13] W. Feller and G. E. Forsythe, New matrix transformation for obtaining characteristic vectors, Quart. Appl. Math. **8**, 325–331 (1951).
[14] J. R. Stock, Die mathematischen Grundlagen für die Organisation der elektronischen Rechenmaschine der Eidgenössische Technische Hochschule (Verlag Birkhäuser, Basel, 1956).
[15] J. Greenstadt, A method for finding roots of arbitrary matrices, Math. Tables and Other Aids to Computation **9**, 47–52 (1955).
[16] S. Gershgorin, Abgrenzung der Eigenwerte einer Matrix, Bul. Acad. Sci. USSR, Leningrad, Classe math. [7] 749–754 (1931).
[17] I. H. Wilkinson, The use of iterative methods for finding the latent roots and vectors of matrices, Math. Tables and Other Aids to Computation **9**, 184–191 (1955).
[18] F. L. Bauer, Das Verfahren der Treppeniteration und verwandte Verfahren zur Lösung algebraischer Eigenwertprobleme. Z. angew. Math. u. Phys. **8** (1957).

U. S. GOVERNMENT PRINTING OFFICE: 1958