J. SIAM NUMER. ANAL.
Ser. B, Vol. 2, No. 2
Printed in U.S.A., 1965

CALCULATING THE SINGULAR VALUES AND PSEUDO-INVERSE OF A MATRIX*

G. GOLUB† AND W. KAHAN‡

Abstract. A numerically stable and fairly fast scheme is described to compute the unitary matrices U and V which transform a given matrix A into a diagonal form $\Sigma = U^*AV$, thus exhibiting A's singular values on Σ's diagonal. The scheme first transforms A to a bidiagonal matrix J, then diagonalizes J. The scheme described here is complicated but does not suffer from the computational difficulties which occasionally afflict some previously known methods. Some applications are mentioned, in particular the use of the pseudo-inverse $A^I = V\Sigma^I U^*$ to solve least squares problems in a way which dampens spurious oscillation and cancellation.

1. Introduction. This paper is concerned with a numerically stable and fairly fast method for obtaining the following decomposition of a given rectangular matrix A:

$$A = U\Sigma V^*, \tag{1.1}$$

where U and V are unitary matrices and Σ is a rectangular diagonal matrix of the same size as A with nonnegative real diagonal entries. These diagonal elements are called the *singular values* or *principal values* of A; they are the nonnegative square roots of the eigenvalues of A^*A or AA^*.

Some applications of the decomposition (1.1) will be mentioned in this paper. In particular, the pseudo-inverse A^I of A will be represented in the form

$$A^I = V\Sigma^I U^*, \tag{1.2}$$

where Σ^I is obtained from Σ by replacing each positive diagonal entry by its reciprocal. The properties and applications of A^I are described in papers by Greville [15], Penrose [25], [26], and Ben-Israel and Charnes [3]. The pseudo-inverse's main value, both conceptually and practically, is that it provides a solution for the following least-squares problem.

Of all the vectors $\mathbf{x}$ *which minimize the sum of squares* $\| \mathbf{b} - A\mathbf{x} \|^2$, *which is the shortest* (*has the smallest* $\| \mathbf{x} \|^2 = \mathbf{x}^*\mathbf{x}$)?

The solution is $\mathbf{x} = A^I\mathbf{b}$. If there were only one vector $\mathbf{x}$ which minimized

* Received by the editors July 14, 1964. This paper was first presented at the Symposium on Matrix Computations at Gatlinburg, Tennessee, in April, 1964.

† Computation Center, Stanford University, Stanford, California. The work of the first author was in part supported by Contract Nonr 225(37) at Stanford University and by The Boeing Scientific Research Laboratories, Seattle, Washington.

‡ University of Toronto, Toronto, Ontario. The second author wishes to thank the National Research Council (of Canada) for their support of the Institute of Computer Sciences at the University of Toronto.

$\| \mathbf{b} - A\mathbf{x} \|$ we would save a bit of work by using

$$A^I = (A^*A)^{-1}A^*$$

instead of (1.2), and this is what we often try to do. But if A^*A is (nearly) singular then there will be infinitely many vectors $\mathbf{x}$ which (nearly) minimize $\| \mathbf{b} - A\mathbf{x} \|$ and the last formula will have to be modified in a way which takes A's rank into account (cf. [4], [6], [7]). The methods considered in this paper simplify the problem of assigning a rank to A.

In the past the conventional way to determine the rank of A was to convert A to a row-echelon form, e.g.,

$$\begin{pmatrix} x & x & x & x & x & \cdot & \cdot & \cdot \\ 0 & x & x & x & x & \cdot & \cdot & \cdot \\ 0 & 0 & x & x & x & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & 0 & 0 & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & 0 & 0 & \cdot & \cdot & \cdot \end{pmatrix}, \qquad \text{(rank = 3)},$$

in which x's represent nonzero elements and 0's represent zeros. The transformation was accomplished by premultiplying A by a succession either of elementary matrices (cf. [5]) or of unitary matrices (cf. [17]) designed to liquidate the subdiagonal elements of each column in turn. In order to obtain a simple picture like the one above it would have been necessary to perform column-interchanges to ensure that the largest possible numbers were being left on the diagonal (cf. "complete pivoting" as described by Wilkinson [33]). It is certainly possible to arrange that in the row-echelon form of A each row will have its largest element on the diagonal. Consequently the rank of A is just the number r of consecutive nonzero terms on the diagonal of its row-echelon form; all rows after the rth are zero. And Σ, correspondingly, should have just r nonzero singular values on its diagonal.

But in floating-point calculations it may not be so easy to decide whether some number is effectively zero or not. Rather, one will try to determine the rank r by observing whether all rows after the rth are negligible in comparison to the first r, with the expectation that the same will be true of the singular values. Even this criterion is hard to apply, as the following example shows:

$$\begin{pmatrix} 1 & -1 & -1 & -1 & -1 & -1 & \cdot & \cdot & \cdot \\ & 1 & -1 & -1 & -1 & -1 & \cdot & \cdot & \cdot \\ & & 1 & -1 & -1 & -1 & \cdot & \cdot & \cdot \\ & & & 1 & -1 & -1 & \cdot & \cdot & \cdot \\ & & & & 1 & -1 & \cdot & \cdot & \cdot \\ & & \mathbf{0} & & & 1 & \cdot & \cdot & \cdot \\ & & & & & & \cdot & \cdot & \cdot \end{pmatrix}.$$

If this matrix, already in row-echelon form, has a sufficiently large number

of rows and columns, then, although it may not appear to the naked eye to be deficient in rank, it is violently ill-conditioned (it has a very tiny singular value), as can be seen by applying the matrix to the column vector whose elements are, in turn,

$$1, 2^{-1}, 2^{-2}, 2^{-3}, \cdots, 2^{-n}, \cdots .$$

On the other hand, when all the -1's in the matrix are replaced by $+1$'s then the resulting matrix is quite docile. Therefore, it would be very hard to tell, by looking at only the diagonal elements of the row-echelon form, whether or not the original matrix A had a singular value sufficiently small to be deleted during the calculation of A^I. In other words, without looking explicitly at the singular values there seems to be no satisfactory way to assign a rank to A.

The singular values of a matrix A are the nonnegative square roots of the eigenvalues of A^*A or AA^*, whichever has fewer rows and columns (see [1]). But the calculation of A^*A using ordinary floating point arithmetic does serious violence to the smaller singular values as well as to the corresponding eigenvectors which appear in U and V in (1.1). A discussion of these points can be found in a paper by Osborne [24], which also contains a nice proof of the existence of the decomposition (1.1). Since the columns of U are the eigenvectors of AA^* and the columns of V are the eigenvectors of A^*A, there is some possibility that a simple calculation of the decomposition (1.1) could be accomplished by using double-precision arithmetic to deal with A^*A and AA^* directly in some way. Such a scheme would be convenient with a machine like the IBM 7094 which has double-precision hardware. But for most other machines, and especially when a programming language deficient in double-precision facilities is used, the complicated scheme described in this paper seems to be the best we have.

Kogbetliantz [18], Hestenes [16], and Forsythe and Henrici [9] have proposed rotational or Jacobi-type methods for obtaining the decomposition (1.1). Kublanovskaja [19] has suggested a QR-type method. These methods are accurate but are slow in terms of total number of operations.

Our scheme is based upon an idea exploited by Lanczos [20]; the matrix

$$\tilde{A} = \begin{pmatrix} 0 & A \\ A^* & 0 \end{pmatrix}$$

has for its eigenvalues the singular values of A, each appearing with both a positive and a negative sign. The representation $\tilde{A}$ could not be treated directly by a standard eigenvalue-vector program without dealing with the problems which we shall discuss in detail in what follows.

2. A matrix decomposition. In order to facilitate the computation of the singular values and the pseudo-inverse of the complex $m \times n$ matrix A, we

describe a convenient matrix decomposition. We assume throughout our discussion that $m \geqq n$ without any loss of generality.

THEOREM 1. *Let A be any $m \times n$ matrix with complex elements. Then A can be decomposed as*

$$A = PJQ^*,$$

where P and Q are unitary matrices and J is an $m \times n$ bidiagonal matrix of the form

$$J = \begin{pmatrix} \alpha_1 & \beta_1 & 0 & \cdot & \cdot & \cdot & 0 \\ & \alpha_2 & \beta_2 & 0 & \cdot & \cdot & \cdot \\ & 0 & \cdot & \cdot & \cdot & & \\ & & & \cdot & \cdot & \cdot & \\ & & & & \cdot & \cdot & \\ & & & & & \cdot & \beta_{n-1} \\ & & & & & & \alpha_n \\ & & & 0 & & & \end{pmatrix} \left.\vphantom{\begin{matrix}a\\a\\a\\a\\a\\a\\a\\a\end{matrix}}\right\} (m-n) \times n$$

Proof. The proof will be a constructive one in which Householder transformations (see [17], [21], [32]) are used. Let $A = A^{(1)}$ and let $A^{(3/2)}$, $A^{(2)}, \cdots, A^{(n)}, A^{(n+1/2)}$ be defined as follows:

$$A^{(k+1/2)} = P^{(k)}A^{(k)}, \qquad k = 1, 2, \cdots, n,$$

$$A^{(k+1)} = A^{(k+1/2)}Q^{(k)}, \qquad k = 1, 2, \cdots, n-1.$$

$P^{(k)}$ and $Q^{(k)}$ are hermitian, unitary matrices of the form

$$P^{(k)} = I - 2\mathbf{x}^{(k)}\mathbf{x}^{(k)*}, \qquad \mathbf{x}^{(k)*}\mathbf{x}^{(k)} = 1,$$

$$Q^{(k)} = I - 2\mathbf{y}^{(k)}\mathbf{y}^{(k)*}, \qquad \mathbf{y}^{(k)*}\mathbf{y}^{(k)} = 1.$$

The unitary transformation $P^{(k)}$ is determined so that

$$a_{i,k}^{(k+1/2)} = 0, \qquad i = k+1, \cdots, m,$$

and $Q^{(k)}$ is determined so that

$$a_{k,j}^{(k+1)} = 0, \qquad j = k+2, \cdots, n,$$

and $A^{(k+1)}$ has the form

$$A^{(k+1)} = \begin{pmatrix} \alpha_1 & \beta_1 & 0 & \cdot & \cdot & & & & & \\ 0 & \alpha_2 & \beta_2 & 0 & \cdot & & & & & \\ \cdot & 0 & \cdot & \cdot & \cdot & & & 0 & & \\ \cdot & \cdot & \cdot & \cdot & \cdot & & & & & \\ & & & & \alpha_k & \beta_k & & & & \\ & & & & & x & x & \cdot & \cdot & \cdot \\ & & 0 & & & x & x & \cdot & \cdot & \cdot \\ & & & & & \cdot & \cdot & \cdot & \cdot & \cdot \\ & & & & & \cdot & \cdot & \cdot & \cdot & \cdot \end{pmatrix}.$$

We illustrate the derivation of the formula for $P^{(k)}$. In order not to disturb those elements which have already been annihilated we set

$$x_i^{(k)} = 0, \qquad i = 1, 2, \cdots, k-1.$$

Since $P^{(k)}$ is a unitary transformation, length is preserved and consequently

$$|\alpha_k|^2 = \sum_{i=k}^{m} |a_{i,k}^{(k)}|^2. \tag{2.1}$$

Also, since $P^{(k)}$ is hermitian,

$$P^{(k)}A^{(k+1/2)} = A^{(k)},$$

so that

$$\begin{aligned}(1 - 2|x_k^{(k)}|^2)\alpha_k &= a_{k,k}^{(k)}, \\ -2x_i^{(k)}\bar{x}_k^{(k)}\alpha_k &= a_{i,k}^{(k)}, \qquad i = k+1, \cdots, m,\end{aligned}$$

and hence

$$|x_k^{(k)}|^2 = \frac{1}{2}\left(1 - \frac{a_{k,k}^{(k)}}{\alpha_k}\right), \tag{2.2}$$

$$x_i^{(k)} = \frac{-a_{i,k}^{(k)}}{2\alpha_k \bar{x}_k^{(k)}}. \tag{2.3}$$

Equations (2.1), (2.2), and (2.3) define two possible vectors $\mathbf{x}^{(k)}$ to within scalar factors of modulus one. In the interest of numerical stability, let us choose sgn α_k so that $x_k^{(k)}$ is as large as possible. Thus

$$\alpha_k = -\frac{a_{k,k}^{(k)}}{|a_{k,k}^{(k)}|}\left(\sum_{i=k}^{m} |a_{i,k}^{(k)}|^2\right)^{1/2}.$$

Summarizing, we have

$$A^{(k+1/2)} = A^{(k)} - \mathbf{x}^{(k)} \cdot 2(\mathbf{x}^{(k)*}A^{(k)}),$$

with

$$\begin{aligned}s_k &= \left(\sum_{i=k}^{m} |a_{i,k}^{(k)}|^2\right)^{1/2}, \\ \alpha_k &= -s_k\left(\frac{a_{k,k}^{(k)}}{|a_{k,k}^{(k)}|}\right), \\ x_i^{(k)} &= 0 \quad \text{for} \quad i < k, \\ x_k^{(k)} &= \left[\frac{1}{2}\left(1 + \frac{|a_{k,k}^{(k)}|}{s_k}\right)\right]^{1/2}, \quad \text{(say)}, \\ c_k &= \left(2s_k \frac{a_{k,k}^{(k)}}{|a_{k,k}^{(k)}|} x_k^{(k)}\right)^{-1},\end{aligned}$$

and

$$x_i^{(k)} = c_k a_{i,k}^{(k)} \quad \text{for} \quad i > k.$$

(If $s_k = 0$, just set $\alpha_k = 0$ and $\mathbf{x}^{(k)} = \mathbf{0}$.) Similarly,

$$A^{(k+1)} = A^{(k+1/2)} - 2(A^{(k+1/2)}\mathbf{y}^{(k)})\cdot\mathbf{y}^{(k)*},$$

with

$$t_k = \left(\sum_{j=k+1}^{n} | a_{k,j}^{(k+1/2)} |^2\right)^{1/2},$$

$$\beta_k = -t_k \cdot \frac{a_{k,k+1}^{(k+1/2)}}{| a_{k,k+1}^{(k+1/2)} |},$$

$$y_j^{(k)} = 0 \quad \text{for} \quad j \leqq k,$$

$$y_{k+1}^{(k)} = \left[\frac{1}{2}\left(1 + \frac{| a_{k,k+1}^{(k+1/2)} |}{t_k}\right)\right]^{1/2}, \quad \text{(say)},$$

$$d_k = \left(2t_k \frac{a_{k,k+1}^{(k+1/2)}}{| a_{k,k+1}^{(k+1/2)} |} y_{k+1}^{(k)}\right)^{-1},$$

and

$$y_j^{(k)} = d_k \bar{a}_{k,j} \quad \text{for} \quad j > k + 1.$$

An alternative approach to bidiagonalizing A is to generate the columns of P and Q sequentially as is done by the Lanczos algorithm for tridiagonalizing a symmetric matrix. The equations

$$AQ = PJ \quad \text{and} \quad P^*A = JQ^*$$

can be expanded in terms of the columns $\mathbf{p}_i$ of P and $\mathbf{q}_i$ of Q to yield

$$\left.\begin{aligned} A\mathbf{q}_1 &= \alpha_1\mathbf{p}_1, \\ A\mathbf{q}_i &= \beta_{i-1}\mathbf{p}_{i-1} + \alpha_i\mathbf{p}_i, \\ \mathbf{p}_{i-1}^*A &= \alpha_{i-1}\mathbf{q}_{i-1}^* + \beta_{i-1}\mathbf{q}_i^*, \\ \mathbf{p}_n^*A &= \alpha_n\mathbf{q}_n^*. \end{aligned}\right\} \qquad i = 2, 3, \cdots, n,$$

These lead to the following algorithm.

(2.4) *Choose* $\mathbf{q}_1$ *arbitrarily with* $\| \mathbf{q}_1 \| = 1$; *then set* $\mathbf{w}_1 = A\mathbf{q}_1$; $\alpha_1 = \| \mathbf{w}_1 \|$, $\mathbf{p}_1 = (\alpha_1)^{-1}\mathbf{w}_1$. *Set* $\mathbf{z}_i^* = \mathbf{p}_i^*A - \alpha_i\mathbf{q}_i^*$, $\beta_i = \| \mathbf{z}_i \|$, $\mathbf{q}_{i+1}^* = (\beta_i)^{-1}\mathbf{z}_i$ *for* $i = 1, 2, \cdots, n-1$; set $\mathbf{w}_i = A\mathbf{q}_i - \beta_{i-1}\mathbf{p}_{i-1}$, $\alpha_i = \| \mathbf{w}_i \|$, $\mathbf{p}_i = (\alpha_i)^{-1}\mathbf{w}_i$ *for* $i = 2, \cdots, n$.

Of course if α_k (β_k) equals zero, one must choose a new vector $\mathbf{p}_k$ ($\mathbf{q}_k$) which is orthogonal to the previously computed $\mathbf{p}_i$'s ($\mathbf{q}_i$'s). It is easy to show then by an inductive proof that the $\mathbf{p}_i$'s and $\mathbf{q}_i$'s generated by (2.4) are the first n columns of the desired unitary matrices P and Q.

Unless an α_k or β_k vanishes, the vector $\mathbf{q}_1$ will completely determine the rest of the vectors $\mathbf{p}_i$ and $\mathbf{q}_i$. Consequently $\mathbf{q}_1$ could be so chosen that the Lanczos-type algorithm would be mathematically identical to the Householder-type algorithm except for a diagonal unitary similarity transformation. But the Lanczos-type algorithm is unstable in the presence of rounding error unless reorthogonalization along the lines suggested by Wilkinson [30] is used. That is, one must restore the orthogonality of the generated vectors by using the Gram-Schmidt method to reorthogonalize each newly generated vector $\mathbf{p}_i$ or $\mathbf{q}_i$ to the previously generated vectors $\mathbf{p}_i$ or $\mathbf{q}_i$, respectively. With the extra work involved in this reorthogonalization, the Lanczos-type algorithm is noticeably slower than the previously described Householder algorithm except possibly if A is a sparse matrix.

3. Computation of the singular values. The singular values of A and of J are the same; they are the positive square roots of J^*J. Let them be called, in order,

$$\sigma_1 \geqq \sigma_2 \geqq \cdots \geqq \sigma_n, \quad \text{where} \quad \sigma_n \geqq 0.$$

These are the numbers which appear on the diagonal of the matrix Σ which was introduced in (1.1), i.e.,

$$\Sigma = \begin{pmatrix} \sigma_1 & & & & \\ & \sigma_2 & & \mathbf{0} & \\ & & \sigma_3 & & \\ & \mathbf{0} & & \ddots & \\ & & & & \sigma_n \\ & & & \mathbf{0} & \end{pmatrix} \Bigg\} (m - n) \times n$$

Analogous to (1.1) is the decomposition

$$J = X\Sigma Y^* \tag{3.1}$$

in which X and Y are unitary matrices which, when they have been calculated, will lead via Theorem 1, $A = PJQ^*$, to the desired decomposition (1.1), namely,

$$A = U\Sigma V^*,$$

with $U = PX$, $V = QY$.

Evidently the last $m - n$ rows of zeros in J do not contribute to the singular values, nor do they have more than a trivial effect upon the determination of X and Y. Therefore it is convenient to delete J's last $m - n$ rows and write

$$J = \begin{pmatrix} \alpha_1 & \beta_1 & & & & & & \\ & \alpha_2 & \beta_2 & & & & & \\ & & \alpha_3 & \beta_3 & & & & \\ & & & \cdot & \cdot & \cdot & & \\ & & & & & \cdot & \cdot & \cdot & \beta_{n-1} \\ & \mathbf{0} & & & & & & \cdot & \alpha_n \end{pmatrix}$$

without introducing any new notation to distinguish this $n \times n$ matrix J from the $m \times n$ matrix J. This can be done because the previous equations remain valid after the following process of "abbreviation":

(i) delete the last $m - n$ rows of zeros in J and Σ;

(ii) delete the last $m - n$ columns of P and U;

(iii) delete the last $m - n$ rows and columns of X; these coincide with the last rows and columns of an $m \times m$ unit matrix. In this section and the next we deal only with the abbreviated matrices.

The singular values σ_i of J are known (cf. [20]) to be related to the eigenvalues of the $2n \times 2n$ matrix

$$\tilde{J} = \begin{pmatrix} 0 & J \\ J^* & 0 \end{pmatrix},$$

whose eigenvalues are just $+\sigma_i$ and $-\sigma_i$ for $i = 1, 2, \cdots, n$. The calculation of the eigenvalues of $\tilde{J}$ is simplified conceptually by a transformation to tridiagonal form via a permutation similarity which will be exhibited now.

Consider the matrix equation

$$\begin{pmatrix} 0 & J \\ J^* & 0 \end{pmatrix} \cdot \begin{pmatrix} \mathbf{x} \\ \pm\mathbf{y} \end{pmatrix} = \pm\sigma \begin{pmatrix} \mathbf{x} \\ \pm\mathbf{y} \end{pmatrix}, \tag{3.2}$$

which, when expanded, takes the form

$$J\mathbf{y} = \sigma\mathbf{x}, \qquad J^*\mathbf{x} = \sigma\mathbf{y},$$

that is, $\alpha_i y_i + \beta_i y_{i+1} = \sigma x_i$, $\bar{\alpha}_1 x_1 = \sigma y_1$, $\alpha_n y_n = \sigma x_n$, $\bar{\beta}_{i-1} x_{i-1} + \bar{\alpha}_i x_i = \sigma y_i$. Now the substitution

$$z_{2i} = x_i, \qquad z_{2i-1} = \pm y_i,$$

leads to the equation

$$T\mathbf{z} = \pm\sigma\mathbf{z}$$

in which T is a $2n \times 2n$ tridiagonal matrix

$$T = \begin{pmatrix} 0 & \bar{\alpha}_1 & & & & & \\ \alpha_1 & 0 & \beta_1 & & & & \\ & \bar{\beta}_1 & 0 & \bar{\alpha}_2 & & & \\ & & \alpha_2 & \cdot & \cdot & & \\ & & & \cdot & \cdot & \cdot & \\ & & 0 & & \cdot & \cdot & \bar{\alpha}_n \\ & & & & & \alpha_n & 0 \end{pmatrix}.$$

Clearly there exists a unitary diagonal matrix D such that the similarity transformation

$$DTD^* = S = \begin{pmatrix} 0 & s_1 & & & & & \\ s_1 & 0 & t_1 & & & & \\ & t_1 & 0 & s_2 & & & \\ & & s_2 & \cdot & \cdot & & \\ & & & \cdot & \cdot & \cdot & \\ & & 0 & & \cdot & \cdot & s_n \\ & & & & & s_n & 0 \end{pmatrix} \tag{3.3}$$

yields a tridiagonal matrix S whose elements

$$s_i = |\alpha_i| \quad \text{and} \quad t_i = |\beta_i|$$

are all real and nonnegative.

There are a number of methods for obtaining the eigenvalues of a tridiagonal symmetric matrix. One of the most accurate and effective methods is to use Sturm sequences; an ALGOL program is given by Wilkinson [35]. One can simplify the algorithm, of course, by taking advantage of the fact that the diagonal elements of T are zero.

Another method of computing the singular values of J is to compute the eigenvalues of

$$J^*J = \begin{pmatrix} |\alpha_1|^2 & \bar{\alpha}_1\beta_1 & & & & & \\ \alpha_1\bar{\beta}_1 & |\alpha_2|^2 + |\beta_1|^2 & \bar{\alpha}_2\beta_2 & & & & \\ & \alpha_2\bar{\beta}_2 & \cdot & \cdot & & & \\ & & \cdot & \cdot & \cdot & & \\ & & & \cdot & \cdot & \cdot & \\ & & 0 & & \cdot & \cdot & \bar{\alpha}_{n-1}\beta_{n-1} \\ & & & & & \alpha_{n-1}\bar{\beta}_{n-1} & |\alpha_n|^2 + |\beta_{n-1}|^2 \end{pmatrix}.$$

Note again that since J^*J is a tridiagonal hermitian matrix there exists a

diagonal unitary matrix Δ such that

$$\Delta(J^*J)\Delta^* = K = \begin{pmatrix} s_1^2 & s_1 t_1 & & & & & \\ s_1 t_1 & s_2^2 + t_1^2 & s_2 t_2 & & & & 0 \\ & s_2 t_2 & \cdot & \cdot & & & \\ & & \cdot & \cdot & \cdot & & \\ & 0 & & \cdot & \cdot & \cdot & \\ & & & & \cdot & \cdot & s_{n-1} t_{n-1} \\ & & & & & s_{n-1} t_{n-1} & s_n^2 + t_{n-1}^2 \end{pmatrix},$$

where $s_i = |\alpha_i|$ and $t_i = |\beta_i|$. Hence K is a real, symmetric, positive semi-definite, tridiagonal matrix and its eigenvalues can be computed by the Sturm sequence algorithm.

Although the smaller eigenvalues of A^*A are usually poorly determined, a simple error analysis shows that all the eigenvalues of K are as well-determined as those of T. The reason for this is that the computation of the Sturm sequences is algebraically the same for both T and K. Thus to use K is preferable since the total number of operations in calculating its eigenvalues is certainly less than in computing the eigenvalues of T.

4. Orthogonal vectors properly paired. We consider now the calculation of the unitary matrices X and Y which were introduced in (3.1):

$$J = X\Sigma Y^*.$$

As pointed out in §3, J can be transformed into a real matrix by means of unitary diagonal transformations, and we shall assume henceforth that this has been done (cf. (3.3)).

To each singular value σ_i corresponds a column $\mathbf{x}_i$ of X and $\mathbf{y}_i$ of Y satisfying

$$J\mathbf{y}_i = \sigma_i \mathbf{x}_i, \qquad J^t\mathbf{x}_i = \sigma_i \mathbf{y}_i. \tag{4.1}$$

Since $J^tJ\mathbf{y}_i = \sigma_i^2\mathbf{y}_i$ one could, in principle, calculate $\mathbf{y}_i$ as the normalized eigenvector of J^tJ corresponding to the eigenvalue σ_i^2, and $\mathbf{x}_i$ could be obtained fom the vector $J\mathbf{y}_i$ either by dividing it by σ_i or by normalizing it. However, if σ_i is small but not quite negligible, then $J\mathbf{y}_i$ will be so much contaminated by the roundoff errors left over after cancellation that the calculated $\mathbf{x}_i$ may well be neither normalized nor orthogonal to the previously calculated $\mathbf{x}$'s.

Another way to calculate X and Y might be to obtain the eigenvectors $\mathbf{x}_i$ and $\mathbf{y}_i$ of JJ^t and J^tJ independently, and then to order the vectors according to the ordering of the corresponding singular values σ_i. But if some of the singular values are too close together then the equations (4.1) are unlikely to be satisfied.

A third way, which seems at first to be free from the objections in the two preceding paragraphs, is to obtain the eigenvectors $\mathbf{z}_i$ of the $2n \times 2n$ tridiagonal matrix S of (3.3). Then the odd-numbered components of $\mathbf{z}_i$ would constitute a vector $\mathbf{y}_i$ and the even-numbered components a vector $\mathbf{x}_i$ which would satisfy (4.1). But in practice trouble shows up here in two ways. First, the facts that (4.1) is very nearly satisfied and that $\mathbf{z}_i$ has been normalized so that $\mathbf{z}_i{}^t\mathbf{z}_i = 2$ do not, in practice, though they should in theory, ensure that $\mathbf{x}_i{}^t\mathbf{x}_i = \mathbf{y}_i{}^t\mathbf{y}_i = 1$. Fortunately, unless σ_i is nearly negligible, one can normalize $\mathbf{x}_i$ and $\mathbf{y}_i$ separately without causing serious extra error. And if σ_i is negligible one can find $\mathbf{x}_i$ and $\mathbf{y}_i$ separately and ensure that they are normalized. The claims in the last two sentences can be proved, but there is no point in doing so because the second source of trouble is more drastic; if the $\mathbf{z}_i$'s are not orthogonal then neither will the $\mathbf{x}_i$'s be orthogonal, nor will the $\mathbf{y}_i$'s. The problem of ensuring that the $\mathbf{z}_i$'s are orthogonal is, in the present state of the art of computation, a serious one.

One way to ensure the orthogonality of calculated eigenvectors of a symmetric matrix is to use Jacobi's method [13], but this is slow. Another way is to reorthogonalize the calculated eigenvectors obtained, say, by inverse iteration with a tridiagonal matrix (cf. [30]); but the extra work done here is no guarantee that the vectors after orthogonalization will still be acceptable as eigenvectors. A third method, and one which seems very promising, involves the use of deflation to "remove" each eigenvector as it is obtained and thereby ensure orthogonality. We shall digress to discuss deflation methods suitable for use with symmetric tridiagonal matrices, and then adapt them to our bidiagonal matrix.

In this digression let K be some real symmetric tridiagonal matrix,

$$K = \begin{pmatrix} a_1 & b_1 & & & & & \\ b_1 & a_1 & b_2 & & & & 0 \\ & b_2 & \cdot & \cdot & & & \\ & & \cdot & \cdot & \cdot & & \\ & & & \cdot & \cdot & \cdot & \\ & 0 & & & \cdot & \cdot & b_{n-1} \\ & & & & b_{n-1} & a_n & \end{pmatrix},$$

of which we already know an eigenvalue λ and its eigenvector $\mathbf{v}$. Rutishauser [27] shows how, in principle, to construct an orthogonal Hessenberg matrix H from the vector $\mathbf{v}$ so that $K_1 = H^tKH$ will have zero in place of b_{n-1}. After deleting the last row and column of the tridiagonal matrix K_1, another eigenvalue, eigenvector and deflation would be calculated, and so on. The eigenvectors of K would be the columns of an orthogonal matrix obtained by multiplying together all the H's. The orthogonality of the eigenvectors would be guaranteed (to within the limits of acceptable rounding

error) irrespective of the closeness of some eigenvalues of K. Rutishauser's method needs some modification because, as Wilkinson [34, p. 189] has shown, the effect of rounding errors in the transformation $K_1 = H^tKH$ could destroy K's tridiagonal form if $\mathbf{v}$'s first few components were too small.

In Rutishauser's deflation the matrix H can be interpreted as a product of 2×2 Jacobi-like rotations applied in succession to the first and second, second and third, third and fourth, $\cdots$, $(n-1)$th and nth rows and columns of K. After the first rotation, each rotation is chosen to annihilate a spurious term which was introduced by the previous rotation. For example, an asterisk in the following figure marks the spurious term which the third rotation must annihilate:

$$\begin{pmatrix} x & x & & & & & \\ x & x & x & * & & & 0 \\ & x & x & x & & & \\ & * & x & x & x & & \\ & & & x & x & x & \\ & & & & \cdot & \cdot & \cdot \\ & & & & & \cdot & \cdot & \cdot \\ & 0 & & & & \cdot & \cdot & \cdot \end{pmatrix}.$$

The first rotation, which fixes all the subsequent ones, can be determined from the first two elements of K's eigenvector $\mathbf{v}$ as suggested by Rutishauser [28, p. 226] or else from the first two elements of $K - \lambda I$. In effect, the deflation of the tridiagonal matrix K is equivalent to a QR-transformation applied to $K - \lambda I$ in the manner suggested by Ortega and Kaiser [22]. Unfortunately, this method also can be shown to be numerically unsatisfactory whenever $\mathbf{v}$'s last few components are abnormally small, because then the element in K_1 which replaces b_{n-1} in K remains too large, in general, to be ignored. Wilkinson [34, p. 187] hints at another method analogous to the one he described in [31, pp. 351–353]; we shall outline this method briefly because we believe it to be an effective compromise between Rutishauser's two schemes.

Having found an eigenvalue λ of K we calculate the corresponding eigenvector $\mathbf{v}$ and normalize it so that its largest component lies between $\frac{1}{2}$ and 2, say. The calculation of $\mathbf{v}$ can be accomplished using the inverse iteration described by Wilkinson [30]; but since there is no way to prove that, in general, one of his inverse iterations will suffice to provide an adequately accurate $\mathbf{v}$, we describe the following method whose properties can be established rigorously. We require that λ be chosen to be the algebraically greatest or else the least eigenvalue of K; this is no inconvenience since in the course of K's successive deflations each of its eigenvalues will be at some time the greatest or the smallest of the current matrix on hand. Next we apply

Gaussian elimination to $K - \lambda I$ *without* pivotal interchanges; there will be no trouble here (cf. [33, pp. 285–286]) provided floating point calculation is used and provided λ, if not exactly right, is larger than K's largest or smaller than K's smallest eigenvalue by perhaps a unit or so in λ's last place. The point here is that each nonzero pivot u_i in the elimination process must be of the same sign as $(K - \lambda I)$'s diagonal elements. The result of the elimination process is to express $K - \lambda I = LU$, where

$$L = \begin{pmatrix} 1 & & & & & \\ l_1 & 1 & & & & \\ & l_2 & 1 & & & 0 \\ & & l_3 & \cdot & & \\ & & & \cdot & \cdot & \\ & 0 & & & \cdot & \cdot \\ & & & & l_{n-1} & 1 \end{pmatrix}$$

and

$$U = \begin{pmatrix} u_1 & b_1 & & & & \\ & u_2 & b_2 & & & 0 \\ & & u_3 & \cdot & & \\ & & & \cdot & \cdot & \\ & & & & \cdot & b_{n-1} \\ & 0 & & & & u_n \end{pmatrix}.$$

Here $u_1 = a_1 - \lambda$ and $l_i = b_i/u_i$, $u_{i+1} = a_{i+1} - \lambda - l_i b_i$ for $i = 1, 2, \cdots, n - 1$. Next we attempt the solution of $(K - \lambda I)\mathbf{v} = \mathbf{r}$ using for $\mathbf{r}$ a vector whose elements all have the same magnitude but signs chosen to maximize the elements of $\mathbf{v}$. The choice of sign is accomplished by first solving $L\mathbf{s} = \mathbf{r}$ as follows:

$$s_1 = +1,$$

$$s_{i+1} = (-l_i s_i) + \operatorname{sgn}(-l_i s_i), \quad i = 1, 2, \cdots, n - 1.$$

The solution of $U\mathbf{v} = \mathbf{s}$ for $\mathbf{v}$ and the subsequent normalization of $\mathbf{v}$ complete the calculation. Provided no two pivots u_i have opposite signs one can show that the elements of $\mathbf{v}$ each have the same signs as the corresponding elements of the desired eigenvector despite the rounding errors committed during $\mathbf{v}$'s calculation. Furthermore, the elements of $\mathbf{r}$ exhibit the same signs as those of $+\mathbf{v}$ or $-\mathbf{v}$, depending upon the sign of the u_i's. Consequently the cosine of the angle between $\mathbf{r}$ and the correct eigenvector is at least $N^{-1/2}$ in magnitude, and finally we conclude that $K\mathbf{v}$ must differ from $\lambda\mathbf{v}$ by no more than a few units in the last place (cf. the argument in [30]). Now even if $\mathbf{v}$ is contaminated by components of the eigenvectors corresponding to other

eigenvalues pathologically close to λ, it will look enough like a true eigenvector to permit the deflation process to proceed. This process for calculating $\mathbf{v}$ is simpler and a little faster than Wilkinson's.

Now that we have $\mathbf{v}$ we proceed to the deflation along the lines outlined by Wilkinson. Each 2×2 rotation is embedded in an $n \times n$ matrix

$$P_j = \begin{pmatrix} 1 & & & & & & & & \\ & 1 & & & & & & & \\ & & \ddots & & & & & & \\ & & & 1 & & & & & \\ & & & & c_j & s_j & & & \\ & & & & -s_j & c_j & & & \\ & & & & & & 1 & & \\ & & & & & & & 1 & \\ & & & & & & & & \ddots \\ & & & & & & & & & 1 \end{pmatrix}, \quad j = 1, 2, \cdots, n-1,$$

with c_j for its jth and $(j+1)$th diagonal elements, where $c_j^2 + s_j^2 = 1$. Suppose the products

$$P_j P_{j-1} \cdots P_1 (K - \lambda I) P_1^t \cdots P_{j-1}^t P_j^t \quad \text{and} \quad P_j P_{j-1} \cdots P_1 \mathbf{v}$$

have the forms

$$P_j\, P_{j-1} \cdots P_1 (K - \lambda I) P_1^t \cdots P_{j-1}^t\, P_j^t = \begin{pmatrix} x & x & & & & & \\ x & x & x & & & & \\ & x & x & h_j & w_j & & \\ & & h_j & x & x & & \\ & & w_j & x & x & x & \\ & & & & x & x & x \\ & & & & & x & x \end{pmatrix}$$

and

$$P_j\, P_{j-1} \cdots P_1\, \mathbf{v} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \phi_{j+1} \\ v_{j+2} \\ x \\ x \end{pmatrix}.$$

At the start we can take

$$h_0 = a_1 - \lambda, \qquad w_0 = b_1, \qquad \phi_1 = v_1.$$

To continue the deflation we must so determine P_{j+1} that its application will simultaneously annihilate the spurious element w_j in the jth row and column of the matrix as well as the vector's $(j+1)$th element ϕ_{j+1}. But in practice the accumulation of rounding errors will prevent the exact annihilation of both elements; instead we shall have to be satisfied with a P_{j+1} which leaves negligible residuals in place of w_j and ϕ_{j+1}. Wilkinson, having scaled $K - \lambda I$ so that its largest element lies between $\frac{1}{2}$ and 2, would use whichever of the equations

$$w_j c_{j+1} = h_j s_{j+1}, \qquad \phi_{j+1} c_{j+1} = -v_{j+2} s_{j+1},$$

contained the largest coefficient $| w_j |$, $| h_j |$, $| \phi_{j+1} |$, or $| v_{j+2} |$ to determine, in conjunction with $c_{j+1}^2 + s_{j+1}^2 = 1$, the values c_{j+1} and s_{j+1}. This method seems to be effective and we believe that it should always work, but since we cannot prove the method's infallibility, our work is incomplete.

Now we can show how to construct a deflation process for the bidiagonal matrix J. The first step is to obtain J's largest singular value σ; σ^2 is the largest eigenvalue of the tridiagonal matrix J^tJ (see §3). The next step requires the corresponding vectors $\mathbf{x}$ and $\mathbf{y}$ which can be obtained either by solving $J^tJ\mathbf{y} = \sigma^2\mathbf{y}$ for $\mathbf{y}$ and setting $\mathbf{x} = \sigma^{-1}J\mathbf{y}$, or by calculating σ's eigenvector $\mathbf{z}$ of S in (3.3) and hence obtaining $\mathbf{x}$ and $\mathbf{y}$ from $\mathbf{z}$'s even and odd components respectively. Both methods for getting $\mathbf{x}$ and $\mathbf{y}$ are numerically stable when performed in floating point. The deflation of J is accomplished by a sequence of 2×2 rotations applied in succession to its first and second columns, its first and second rows, its second and third columns, its second and third rows, its third and fourth columns, $\cdots$, its $(n-1)$th and nth rows. The ith rotation applied to rows i and $i+1$ of J must simultaneously annihilate a spurious subdiagonal element, introduced into row $i+1$ by the previous column rotation, and the ith element in the current $\mathbf{x}$-vector. The ith column rotation, except for the first, must annihilate a spurious term introduced by the previous row rotation into the $(i+1)$th column just above the first superdiagonal, and simultaneously the transpose of the ith column rotation must liquidate the ith element of the current $\mathbf{y}$-vector. The first column rotation would when applied to $J^tJ - \sigma^2 I$ annihilate the element in its first row and second column. At the end of the deflation process J's element b_{n-1} should have been replaced by zero. Of course, rounding errors will prevent the rotations from performing their roles exactly upon both the matrix J and the vectors $\mathbf{x}$ and $\mathbf{y}$, but just as in the deflation of a tridiagonal matrix we are able so to determine the rotations that negligible residuals are left behind in place of the elements we wished to liquidate.

After deflating J we delete its last row and column and repeat the process until J is deflated to a 1×1 matrix or the deflated J becomes negligibly small. At the end we multiply the rotations in reverse order to construct the

matrices X and Y which put J into the form (3.1):

$$J = X\Sigma Y.$$

(If J was complex, a unitary diagonal transformation should be incorporated here.) Finally the matrices P and Q of Theorem 1 are multiplied thus:

$$U = PX, \qquad V = QY,$$

to exhibit the decomposition (1.1):

$$A = U\Sigma V.$$

The two matrix multiplications PX and QY take most of the work.

5. Applications. The basic decomposition given by (1.1) has many applications in data analysis and applied mathematics. Suppose the matrix A arises from statistical observation, and we wish to replace A by another matrix $\hat{A}$ (say) which has lower rank p and is the best approximation to A in some sense. If we use the Frobenius norm (i.e., $\| A \|^2 = \text{trace } A^*A$) then the problem has been solved [8] as follows.

THEOREM 2. *Let A be an $m \times n$ matrix of rank r which has complex elements. Let S_p be the set of all $m \times n$ matrices of rank $p < r$. Then for all $B \in S_p$,*

$$\| A - \hat{A} \| \leqq \| A - B \|,$$

where

$$\hat{A} = U\hat{\Sigma}V^*$$

and $\hat{\Sigma}$ is obtained from the Σ of (1.1) by setting to zero all but its p largest singular values σ_i.

Proof. Since $A = U\Sigma V^*$ and the Frobenius norm is unitarily invariant,

$$\| A - B \| = \| \Sigma - U^*BV \|.$$

Let $U^*BV = C$. Then

$$\| \Sigma - C \|^2 = \sum_{i=1}^{n} | \sigma_i - c_{ii} |^2 + \sum_{i \neq j} | c_{ij} |^2 \geqq \sum_{i=1}^{n} | \sigma_i - c_{ii} |^2.$$

Now it is convenient to order the singular values in such a way that $\sigma_i \geqq \sigma_{i+1}$. Thus, $\| A - B \|^2$ is minimized if $c_{ii} = \sigma_i$ for $i = 1, 2, \cdots, p$, and $c_{ij} = 0$ otherwise, i.e., for $C = \hat{\Sigma}$. Obviously,

$$\| A - \hat{A} \| = (\sigma_{p+1}^2 + \cdots + \sigma_r^2)^{1/2}.$$

Finding the vector $\mathbf{x}$ of shortest length which minimizes $\| \mathbf{b} - A\mathbf{x} \|$ is equivalent to finding the vector $\mathbf{y}$ of shortest length which minimizes

$\| \mathbf{c} - J\mathbf{y} \|$, where $\mathbf{c} = P^*\mathbf{b}$ and $\mathbf{y} = Q^*\mathbf{x}$. Here a natural question arises: is there any method which bypasses the complicated scheme in §3 and §4 for exhibiting J's singular values explicitly, and instead takes advantage of J's simple bidiagonal form to solve the least squares problem or to calculate J^I? Such a method, if it exists, must retain provision for intentional perturbations designed to delete, in effect, negligible singular values without inducing too large a discrepancy in J or A. Unfortunately, J's simple form is deceptive; even J's rank is hard to estimate without further calculation. For example, if J's rank r is less than n, then at least $n - r$ of the α_i's, and possibly more, should vanish; but in practice none of the α_i's may be negligible even though several may be very small compared with adjacent β_i's and, in consequence, a few of J's singular values may be negligible.

Perhaps the recurrence described by Greville [15] can be modified by the introduction of pivoting and then applied to J to calculate J^I. Until this scheme is worked out, the best method we can suggest for solving the least squares problem together with controllable perturbations is the following. Compute explicitly the representation

$$A = U\Sigma V^*,$$

decide which of the singular values are small enough to ignore, replace the remaining singular values by their reciprocals to obtain Σ^I, and finally use

$$A^I = V\Sigma^I U^*$$

to obtain the least squares solution $\mathbf{x} = A^I\mathbf{b}$. Once again, to ignore some singular values $\sigma_{r+1}, \sigma_{r+2}, \cdots, \sigma_n$ is equivalent to perturbing A by a matrix whose norm is $(\sum_{i=r+1}^{n} \sigma_i^2)^{1/2}$.

In some scientific calculations it is preferable that a given square matrix A be perturbed as little as possible (just rounding errors), but instead a perturbation $\delta\mathbf{b}$ in the right-hand side $\mathbf{b}$ of the equation $A\mathbf{x} = \mathbf{b}$ is permissible provided $\| \delta\mathbf{b} \|$ does not exceed a given tolerance ϵ. The substitution

$$\mathbf{y} = V^*\mathbf{x}, \qquad \mathbf{c} = U^*\mathbf{b}, \qquad \delta\mathbf{c} = U^*\delta\mathbf{b},$$

transforms the perturbed equation $A\mathbf{x} = \mathbf{b} + \delta\mathbf{b}$ into an equivalent diagonal system

$$\Sigma\mathbf{y} = \mathbf{c} + \delta\mathbf{c},$$

in which the permissible perturbation $\delta\mathbf{c}$ still satisfies

$$\| \delta\mathbf{c} \| < \epsilon. \tag{5.1}$$

Subject to this constraint, $\delta\mathbf{c}$ may be chosen to optimize some other criterion. For example, suppose we wish to minimize $\| \mathbf{x} \| = \| \mathbf{y} \|$. Then ideally $\delta\mathbf{c}$

should satisfy $\Sigma^2\delta\mathbf{c} + \lambda(\mathbf{c} + \delta\mathbf{c}) = 0$ with some suitable positive value of the Lagrange multiplier λ sufficiently small so that (5.1) is satisfied too. But for most practical purposes it is sufficient to use trial and error to determine λ to within a factor of two so that $\delta\mathbf{c} = -(I + \lambda^{-1}\Sigma^2)^{-1}\mathbf{c}$ will satisfy $\delta\mathbf{c}^*\delta\mathbf{c} < \epsilon^2$. The use of such a technique in least squares problems tends to suppress violent oscillation and cancellation which might otherwise detract from the usefulness of the solution $\mathbf{x}$.

A similar technique is valuable for the solution of the sets of linear equations which approximate integral equations of the form

$$\int A(i,j)x(j)\,dj = b(i).$$

Here the numerical treatment of the integral equation, in using singular values, is similar to the theoretical treatment found in [29]. Once again, the use of the decomposition $A = U\Sigma V^*$ aids the suppression of spurious oscillations in the function x.

We close with a warning; diagonal transformations can change A's singular values and A^I in a nontrivial way. Therefore some sort of equilibration may be necessary to allow each row and column of A to communicate its proper significance to the calculation. Two useful forms of equilibration are:

(i) scale each row and column of A in such a way that all the rows have roughly the same norm and so have all the columns;

(ii) scale each row and column of A in such a way that the absolute uncertainty in each element of A does not vary much from element to element. On least squares problems such equilibration is accomplished by weighting each residual in the sum of squares (see [2], [10], [11], [23] on equilibration algorithms, and [14]).

REFERENCES

[1] A. R. Amir-Moéz and A. L. Fass, *Elements of Linear Spaces*, Pergamon, New York, 1962, Chap. 12.

[2] F. L. Bauer, *Optimally scaled matrices*, Numer. Math., 5 (1963), pp. 73–87.

[3] A. Ben-Israel and A. Charnes, *Contributions to the theory of generalized inverses*, J. Soc. Indust. Appl. Math., 11 (1963), pp. 667–699.

[4] A. Ben-Israel and S. J. Wersan, *An elimination method for computing the generalized inverse of an arbitrary complex matrix*, J. Assoc. Comput. Mach., 10 (1963), pp. 532–537.

[5] G. Birkhoff and S. MacLane, *A Survey of Modern Algebra*, Macmillan, New York, 1953, Chap. VII, §6.

[6] J. W. Blattner, *Bordered matrices*, J. Soc. Indust. Appl. Math., 10 (1962), pp. 528–536.

[7] J. C. G. Boot, *The computation of the generalized inverse of singular or rectangular matrices*, Amer. Math. Monthly, 70 (1963), pp. 302–303.

[8] C. Eckart and G. Young, *The approximation of one matrix by another of lower rank*, Psychometrika, 1 (1936), pp. 211–218.

[9] G. E. Forsythe and P. Henrici, *The cyclic Jacobi method for computing the principal values of a complex matrix*, Trans. Amer. Math. Soc., 94 (1960), pp. 1–23.

[10] G. E. Forsythe and E. G. Straus, *On best conditioned matrices*, Proc. Amer. Math. Soc., 6 (1955), pp. 340–345.

[11] D. R. Fulkerson and P. Wolfe, *An algorithm for scaling matrices*, SIAM Rev., 4 (1962), pp. 142–146.

[12] W. Givens, *Numerical computation of the characteristic values of a real symmetric matrix*. Oak Ridge National Laboratory, Report No. 1574, 1954.

[13] H. H. Goldstine, F. J. Murray, and J. von Neumann, *The Jacobi method for real symmetric matrices*, J. Assoc. Comput. Mach., 6 (1959), pp. 59–96.

[14] G. H. Golub, *Comparison of the variance of minimum variance and weighted least squares regression coefficients*, Ann. Math. Statist., 34 (1963), pp. 984–991.

[15] T. N. E. Greville, *Some applications of the pseudo-inverse of a matrix*, SIAM Rev., 2 (1960), pp. 15–22.

[16] M. R. Hestenes, *Inversion of matrices by biorthogonalization and related results*, J. Soc. Indust. Appl. Math., 6 (1958), pp. 51–90.

[17] A. S. Householder, *Unitary triangularization of a nonsymmetric matrix*, J. Assoc. Comput. Mach., 5 (1958), pp. 339–342.

[18] E. G. Kogbetliantz, *Solution of linear equations by diagonalization of coefficients matrix*, Quart. Appl. Math., 13 (1955), pp. 123–132.

[19] V. N. Kublanovskaja, *Some algorithms for the solution of the complete problem of eigenvalues*, V. Vyčisl. Mat.i. Mat. Fiz., 1 (1961), pp. 555–570.

[20] C. Lanczos, *Linear Differential Operators*, Van Nostrand, London, 1961, Chap. 3.

[21] D. D. Morrison, *Remarks on the unitary triangularization of a nonsymmetric matrix*, J. Assoc. Comput. Mach., 7 (1960), pp. 185–186.

[22] J. M. Ortega and H. F. Kaiser, *The LL^t and QR methods for symmetric tridiagonal matrices*, Comput. J., 6 (1963), pp. 99–101.

[23] E. E. Osborne, *On pre-conditioning of matrices*, J. Assoc. Comput. Mach., 7 (1960), pp. 338–345.

[24] ———, *On least squares solutions of linear equations*, Ibid., 8 (1961), pp. 628–636.

[25] R. Penrose, *A generalized inverse for matrices*, Proc. Cambridge Philos. Soc., 51 (1955), pp. 406–413.

[26] ———, *On best approximate solutions of linear matrix equations*, Ibid., 52 (1956), pp. 17–19.

[27] H. Rutishauser, *Deflation bei Bandmatrizen*, Z. Angew. Math. Phys., 10 (1959), pp. 314–319.

[28] ———, *On Jacobi rotation patterns*, Proc. Symp. Appl. Math. XV, Experimental Arithmetic, High Speed Computing, and Mathematics, Amer. Math. Soc., 1963, pp. 219–240.

[29] F. Smithies, *Integral Equations*, Cambridge University Press, Cambridge, 1958, Chap. VIII.

[30] J. H. Wilkinson, *The calculation of the eigenvectors of codiagonal matrices*, Comput. J., 1 (1958), pp. 148–152.

[31] ———, *Stability of the reduction of a matrix to almost triangular and triangular forms by elementary similarity transformations*, J. Assoc. Comput. Mach., 6 (1959), pp. 336–359.

[32] ———, *Householder's method for the solution of the algebraic eigenproblem*, Comput. J., 3 (1960), pp. 23–27.
[33] ———, *Error analysis of direct methods of matrix inversion*, J. Assoc. Comput. Mach., 8 (1961), pp. 281–330.
[34] ———, *Error analysis of eigenvalue techniques based on orthogonal transformations*, J. Soc. Indust. Appl. Math., 10 (1962), pp. 162–195.
[35] ———, *Calculation of the eigenvalues of a symmetric tridiagonal matrix by the method of bisection*, Numer. Math., 4 (1962), pp. 362–367.