Journal of Research of the National Bureau of Standards Vol. 49, No. 6, December 1952 Research Paper 2379

Methods of Conjugate Gradients for Solving Linear Systems[1]

Magnus R. Hestenes [2] and Eduard Stiefel [3]

An iterative algorithm is given for solving a system $Ax=k$ of n linear equations in n unknowns. The solution is given in n steps. It is shown that this method is a special case of a very general method which also includes Gaussian elimination. These general algorithms are essentially algorithms for finding an n dimensional ellipsoid. Connections are made with the theory of orthogonal polynomials and continued fractions.

1. Introduction

One of the major problems in machine computations is to find an effective method of solving a system of n simultaneous equations in n unknowns, particularly if n is large. There is, of course, no best method for all problems because the goodness of a method depends to some extent upon the particular system to be solved. In judging the goodness of a method for machine computations, one should bear in mind that criteria for a good machine method may be different from those for a hand method. By a hand method, we shall mean one in which a desk calculator may be used. By a machine method, we shall mean one in which sequence-controlled machines are used.

A machine method should have the following properties:

(1) The method should be simple, composed of a repetition of elementary routines requiring a minimum of storage space.

(2) The method should insure rapid convergence if the number of steps required for the solution is infinite. A method which—if no rounding-off errors occur—will yield the solution in a finite number of steps is to be preferred.

(3) The procedure should be stable with respect to rounding-off errors. If needed, a subroutine should be available to insure this stability. It should be possible to diminish rounding-off errors by a repetition of the same routine, starting with the previous result as the new estimate of the solution.

(4) Each step should give information about the solution and should yield a new and better estimate than the previous one.

(5) As many of the original data as possible should be used during each step of the routine. Special properties of the given linear system—such as having many vanishing coefficients—should be preserved. (For example, in the Gauss elimination special properties of this type may be destroyed.)

In our opinion there are two methods that best fit these criteria, namely, (a) the Gauss elimination method; (b) the conjugate gradient method presented in the present monograph.

There are many variations of the elimination method, just as there are many variations of the conjugate gradient method here presented. In the present paper it will be shown that both methods are special cases of a method that we call the method of conjugate directions. This enables one to compare the two methods from a theoretical point of view.

In our opinion, the conjugate gradient method is superior to the elimination method as a machine method. Our reasons can be stated as follows:

(a) Like the Gauss elimination method, the method of conjugate gradients gives the solution in n steps if no rounding-off error occurs.

(b) The conjugate gradient method is simpler to code and requires less storage space.

(c) The given matrix is unaltered during the process, so that a maximum of the original data is used. The advantage of having many zeros in the matrix is preserved. The method is, therefore, especially suited to handle linear systems arising from difference equations approximating boundary value problems.

(d) At each step an estimate of the solution is given, which is an improvement over the one given in the preceding step.

(e) At any step one can start anew by a very simple device, keeping the estimate last obtained as the initial estimate.

In the present paper, the conjugate gradient routines are developed for the symmetric and nonsymmetric cases. The principal results are described in section 3. For most of the theoretical considerations, we restrict ourselves to the positive definite symmetric case. No generality is lost thereby. We deal only with real matrices. The extension to complex matrices is simple.

The method of conjugate gradients was developed independently by E. Stiefel of the Institute of Applied Mathematics at Zurich and by M. R. Hestenes with the cooperation of J. B. Rosser, G. Forsythe, and L. Paige of the Institute for Numerical Analysis, National Bureau of Standards. The present account was prepared jointly by M. R. Hestenes and E. Stiefel during the latter's stay at the National Bureau of Standards. The first papers on this method were

[1] This work was performed on a National Bureau of Standards contract with the University of California at Los Angeles, and was sponsored (in part) by the Office of Naval Research, United States Navy.

[2] National Bureau of Standards and University of California at Los Angeles.

[3] University of California at Los Angeles, and Eidgenössische Technische Hochschule, Zurich, Switzerland.

given by E. Stiefel[4] and by M. R. Hestenes.[5] Reports on this method were given by E. Stiefel[6] and J. B. Rosser[7] at a Symposium[8] on August 23–25, 1951. Recently, C. Lanczos[9] developed a closely related routine based on his earlier paper on eigenvalue problem.[10] Examples and numerical tests of the method have been by R. Hayes, U. Hochstrasser, and M. Stein.

2. Notations and Terminology

Throughout the following pages we shall be concerned with the problem of solving a system of linear equations

$$\begin{aligned} a_{11}x_1+a_{12}x_2+ \ . \ . \ . \ +a_{1n}x_n&=k_1 \\ a_{21}x_1+a_{22}x_2+ \ . \ . \ . \ +a_{2n}x_n&=k_2 \\ . \ . \ . \quad . \ . \ . \quad . \ . \ . & \\ a_{n1}x_1+a_{n2}x_2+ \ . \ . \ . \ +a_{nn}x_n&=k_n. \end{aligned} \tag{2:1}$$

These equations will be written in the vector form $Ax=k$. Here A is the matrix of coefficients (a_{ij}), x and k are the vectors $(x_1,. \ . \ .,x_n)$ and $(k_1,. \ . \ .,k_n)$. It is assumed that A is nonsingular. Its *inverse* A^{-1} therefore exists. We denote the *transpose* of A by A^*.

Given two vectors $x=(x_1,. \ . \ .,x_n)$ and $y=(y_1,. \ . \ .,y_n)$, their sum $x+y$ is the vector $(x_1+y_1,. \ . \ .,x_n+y_n)$, and ax is the vector $(ax_1,. \ . \ .,ax_n)$, where a is a scalar. The sum

$$(x,y)=x_1y_1+x_2y_2+. \ . \ .+x_ny_n$$

is their *scalar product*. The *length* of x will be denoted by

$$|x|=(x_1^2+. \ . \ .+x_n^2)^{\frac{1}{2}}=(x,x)^{\frac{1}{2}}.$$

The *Cauchy-Schwarz inequality* states that for all x,y:

$$(x,y)^2\leq(x,x)(y,y) \quad \text{or} \quad |(x,y)|\leq|x|\,|y|. \tag{2:2}$$

The matrix A and its transpose A^* satisfy the relation

$$(x,Ay)=\sum_{i,j=1}^{n}a_{ij}x_iy_j=(A^*x,y).$$

If $a_{ij}=a_{ji}$, that is, if $A=A^*$, then A is said to be *symmetric*. A matrix A is said to be *positive definite* in case $(x,Ax)>0$ whenever $x\neq0$. If $(x,Ax)\geqq0$ for all x, then A is said to be *nonnegative*. If A is symmetric, then two vectors x and y are said to be *conjugate* or *A-orthogonal* if the relation $(x,Ay)=(Ax,y)=0$ holds. This is an extension of the orthogonality relation $(x,y)=0$.

By an *eigenvalue* of a matrix A is meant a number λ such that $Ay=\lambda y$ has a solution $y\neq0$, and y is called a corresponding *eigenvector*.

Unless otherwise expressly stated the matrix A, with which we are concerned, will be *assumed to be symmetric and positive definite*. Clearly no loss of generality is caused thereby from a theoretical point of view, because the system $Ax=k$ is equivalent to the system $Bx=l$, where $B=A^*A$, $l=A^*k$. From a numerical point of view, the two systems are different, because of rounding-off errors that occur in joining the product A^*A. Our applications to the nonsymmetric case do not involve the computation of A^*A.

In the sequel we shall not have occasion to refer to a particular coordinate of a vector. Accordingly we may use subscripts to distinguish vectors instead of components. Thus x_0 will denote the vector $(x_{01},. \ . \ .,x_{0n})$ and x_i the vector $(x_{i1},. \ . \ .,x_{in})$. In case a symbol is to be interpreted as a component, we shall call attention to this fact unless the interpretation is evident from the context.

The *solution of the system $Ax=k$ will be denoted by h*; that is, $Ah=k$. If x is an estimate of h, the difference $r=k-Ax$ will be called the *residual* of x as an estimate of h. The quantity $|r|^2$ will be called the *sauared residual*. The vector $h-x$ will be called the *error vector* of x, as an estimate of h.

3. Method of Conjugate Gradients (cg-Method)

The present section will be devoted to a description of a method of solving a system of linear equations $Ax=k$. This method will be called the *conjugate gradient method* or, more briefly, the cg-method, for reasons which will unfold from the theory developed in later sections. For the moment, we shall limit ourselves to collecting in one place the basic formulas upon which the method is based and to describing briefly how these formulas are used.

The cg-method is an iterative method which terminates in at most n steps if no rounding-off errors are encountered. Starting with an initial estimate x_0 of the solution h, one determines successively new estimates $x_0, x_1, x_2, . \ . \ .$ of h, the estimate x_i being closer to h than x_{i+1}. At each step the residual $r_i=k-Ax_i$ is computed. Normally this vector can be used as a measure of the "goodness" of the estimate x_i. However, this measure is not a reliable one because, as will be seen in section 18, it is possible to construct cases in which the *squared residual* $|r_i|^2$ increases at each step (except for the last) while the length of the error vector $|h-x_i|$ decreases monotonically. If no rounding-off error is encountered, one will reach an estimate $x_m(m\leqq n)$ at which $r_m=0$. This estimate is the desired solution h. Normally, $m=n$. However, since rounding-

[4] E. Stiefel, Uebereinige Methoden der Relaxationsrechnung, Z. angew. Math. Physik (3) (1952).

[5] M. R. Hestenes, Iterative methods for solving linear equations, NAML Report 52-9, National Bureau of Standards (1951).

[6] E. Stiefel, Some special methods of relaxation techniques, to appear in the Proceedings of the symposium (see footnote 8).

[7] J. B. Rosser, Rapidly converging iterative methods for solving linear equations, to appear in the Proceedings of the symposium (see footnote 8).

[8] Symposium on simultaneous linear equations and the determination of eigenvalues, Institute for Numerical Analysis, National Bureau of Standards, held on the campus of the University of California at Los Angeles (August 23–25, 1951).

[9] C. Lanczos, Solution of systems of linear equations by minimized iterations, NAML Report 52-13, National Bureau of Standards (1951).

[10] C. Lanczos, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, J. Research NBS 45, 255 (1950) RP2133; Proceedings of Second Symposium on Large-Scale Digital Calculating Machinery, Cambridge, Mass., 1951, pages 164–206.

off errors always occur except under very unusual circumstances, the estimate x_n in general will not be the solution h but will be a good approximation of h. If the residual r_n is too large, one may continue with the iteration to obtain better estimates of h. Our experience indicates that frequently x_{n+1} is considerably better than x_n. One should not continue too far beyond x_n but should start anew with the last estimate obtained as the initial estimate, so as to diminish the effects of rounding-off errors. As a matter of fact one can start anew at any step one chooses. This flexibility is one of the principal advantages of the method.

In case the matrix A is *symmetric* and *positive definite*, the following formulas are used in the conjugate gradient method:

$$p_0=r_0=k-Ax_0 \qquad (x_0 \text{ arbitrary}) \tag{3:1a}$$

$$a_i=\frac{|r_i|^2}{(p_i,Ap_i)}, \tag{3:1b}$$

$$x_{i+1}=x_i+a_ip_i, \tag{3:1c}$$

$$r_{i+1}=r_i-a_iAp_i, \tag{3:1d}$$

$$b_i=\frac{|r_{i+1}|^2}{|r_i|^2}, \tag{3:1e}$$

$$p_{i+1}=r_{i+1}+b_ip_i. \tag{3:1f}$$

In place of the formulas (3:1b) and (3:1e) one may use

$$a_i=\frac{(p_i,r_i)}{(p_i,Ap_i)}, \tag{3:2a}$$

$$b_i=-\frac{(r_{i+1},Ap_i)}{(p_i,Ap_i)}. \tag{3:2b}$$

Although these formulas are slightly more complicated than those given in (3:1), they have the advantage that scale factors (introduced to increase accuracy) are more easily changed during the course of the computation.

The *conjugate gradient method* (cg-method) is given by the following steps:

Initial step: Select an estimate x_0 of h and compute the residual r_0 and the direction p_0 by formulas (3:1a).

General routine: Having determined the estimate x_i of h, the residual r_i, and the direction p_i, compute x_{i+1}, r_{i+1}, and p_{i+1} by formulas (3:1b), . . ., (3:1f) successively.

As will be seen in section 5, the residuals r_0, r_1, . . . are mutually orthogonal, and the direction vectors p_0, p_1, . . . are mutually conjugate, that is,

$$(r_i,\ r_j)=0, \quad (p_i,\ Ap_j)=0 \qquad (i\neq j). \tag{3:3}$$

These relations can be used as checks.

Once one has obtained the set of n mutually conjugate vectors p_0, . . ., p_{n-1} the solution of

$$Ax=k' \tag{3:4}$$

can be obtained by the formula

$$x=\sum_{i=0}^{n-1}\frac{(p_i,k')}{(Ap_i,p_i)}\,p_i. \tag{3:5}$$

It follows that, if we denote by p_{ij} the jth component of p_i, then

$$\sum_{i=0}^{n-1}\frac{p_{ij}p_{ik}}{(p_i,Ap_i)}$$

is the element in the jth row and kth column of the inverse A^{-1} of A.

There are two objections to the use of formula (3:5). First, contrary to the procedure of the general routine (3:1), this would require the storage of the vectors p_0, p_1, ... This is impractical, particularly in large systems. Second, the results obtained by this method are much more influenced by rounding-off errors than those obtained by the step-by-step routine (3:1).

In the cg-method *the error vector $h-x$ is diminished in length* at each step. The quantity $f(x)=(h-x,\ A\ (h-x))$, called the *error function*, is also diminished at each step. But the *squared residual* $|r|^2=|k-Ax|^2$ normally oscillates and may even increase. There is a modification of the cg-method where all three quantities diminish at each step. This modification is given in section 7. It has an advantage and a disadvantage. Its disadvantage is that the error vector in each step is longer than in the original method. Moreover, the computation is complicated, since it is a routine superimposed upon the original one. However, in the special case where the given linear equation system arises from a difference approximation of a boundary-value problem, it can be shown that the estimates are smoother in the modified method than in the original. This may be an advantage if the desired solution is to be differentiated afterwards.

Concurrently with the solution of a given linear system, characteristic roots of its matrix may be obtained: compute the values of the polynomials R_0, R_1, . . . and P_0, P_1, . . . at λ by the iteration

$$R_0=P_0=1$$

$$R_{i+1}=R_i-\lambda a_iP_i$$

$$P_{i+1}=R_{i+1}+b_iP_i. \tag{3:6}$$

The last polynomial $R_m(\lambda)$ is a factor of the characteristic polynomial of A and coincides with it when $m=n$. The characteristic roots, which are the zeros of $R_m(\lambda)$, can be found by Newton's methods without actually computing the polynomial $R_m(\lambda)$ itself. One uses the formulas

$$\lambda_{k+1}=\lambda_k-\frac{R_m(\lambda_k)}{R'_m(\lambda_k)}, \tag{3:7}$$

where $R_m(\lambda_k)$, $R'_m(\lambda_k)$ are determined by the iteration (3:6) and,

$$R'_0=P'_0=0$$

$$R'_{i+1}=R'_i-\lambda a_iP'_i-a_iP_i$$

$$P'_{i+1}=R'_i+b_iP'_i$$

with $\lambda=\lambda_k$. In this connection, it is of interest to observe that if $m=n$, the determinant of A is given by the formula

$$\det(A)=\frac{1}{a_0a_1\ .\ .\ .\ a_{n-1}}.$$

The cg-method can be extended to the case in which A is a *general nonsymmetric and nonsingular* matrix. In this case one replaces eq (3:1) by the set

$$\begin{aligned}
&r_0=k-Ax_0, \qquad p_0=A^*r_0,\\
&a_i=\frac{|A^*r_i|^2}{|Ap_i|^2}\\
&x_{i+1}=x_i+a_ip_i,\\
&r_{i+1}=r_i-a_iAp_i,\\
&b_i=\frac{|A^*r_{i+1}|^2}{|A^*r_i|^2}\\
&p_{i+1}=A^*r_{i+1}+b_ip_i.
\end{aligned}\tag{3:8}$$

This system is discussed in section 10.

4. Method of Conjugate Directions (cd-Method)[1]

The cg-method can be considered as a special case of a general method, which we shall call the *method of conjugate directions* or more briefly the cd-method. In this method, the vectors p_0, p_1, . . . are selected to be mutually conjugate but have no further restrictions. It consists of the following routine:

Initial step. Select an estimate x_0 of h (the solution), compute the residual $r_0=k-Ax_0$, and choose a direction p_0.

General routine. Having obtained the estimate x_i of h, the residual $r_i=k-Ax_i$ and the direction p_i, compute the new estimate x_{i+1} and its residual r_{i+1} by the formulas

$$a_i=\frac{(p_i,r_i)}{(p_i,Ap_i)},\tag{4:1a}$$

$$x_{i+1}=x_i+a_ip_i,\tag{4:1b}$$

$$r_{i+1}=r_i-a_iAp_i.\tag{4:1c}$$

Next select a direction p_{i+1} conjugate to p_0, . . ., p_i, that is, such that

$$(p_{i+1},Ap_j)=0 \qquad (j=0,1,\ .\ .\ .\ ,i).\tag{4:2}$$

In a sense the cd-method is not precise, in that no formulas are given for the computation of the directions p_0, p_1, ... Various formulas can be given, each leading to a special method. The formula (3:1f) leads to the cg-method. It will be seen in section 12 that the case in which the p's are obtained by an A-orthogonalization of the basic vectors (1,0, . . ., 0), (0,1,0, . . .), . . . leads essentially to the Gauss elimination method.

The basic properties of the cd-method are given by the following theorems.

Theorem 4:1. *The direction vectors* p_0, p_1, $\cdot\cdot\cdot$ *are mutually conjugate. The residual vector* r_i *is orthogonal to* p_0, p_1, $\cdot\cdot\cdot$, p_{i-1}. *The inner product of* p_i *with each of the residuals* r_0, r_1, $\cdot\cdot\cdot$, r_i *is the same. That is,*

$$(p_i,Ap_j)=0 \qquad (i\neq j)\tag{4:3a}$$

$$(p_j,r_i)=0 \qquad (j=0,1,\ \cdot\cdot\cdot,i-1)\tag{4:3b}$$

$$(p_i,r_0)=(p_i,r_1)=\ \cdot\cdot\cdot\ =(p_i,r_i).\tag{4:3c}$$

The scalar a_i *can be given by the formula*

$$a_i=\frac{(p_i,r_0)}{(p_i,Ap_i)}\tag{4:4}$$

in place of (4:1a).

Equation (4:3a) follows from (4:2). Using (4:1c), we find that

$$(p_j,r_{k+1})=(p_j,r_k)-a_k(p_j,Ap_k).$$

If $j=k$ we have, by (4:1a), $(p_k,r_{k+1})=0$. Moreover, by (4:3a) $(p_j,r_{k+1})=(p_j,r_k)$, $(j\neq k)$. Equations (4:3b) and (4:3c) follow from these relations. The formula (4:4) follows from (4:3c) and (4:1a).

As a consequence of (4:4) the estimates x_1,x_2, $\cdot\cdot\cdot$ of h can be computed without computing the residuals r_0,r_1, $\cdot\cdot\cdot$, provided that the choice of the direction vectors p_0,p_1, $\cdot\cdot\cdot$ is independent of these residuals.

Theorem 4:2. *The cd-method is an m-step method* ($m\leq n$) *in the sense that at the mth step the estimate* x_m *is the desired solution h.*

For let m be the first integer such that $y_0=h-x_0$ is in the subspace spanned by p_0, $\cdot\cdot\cdot$, p_{m-1}. Clearly, $m\leq n$, since the vectors p_0,p_1, $\cdot\cdot\cdot$ are linearly independent. We may, accordingly, choose scalars α_0, $\cdot\cdot\cdot$, α_{m-1} such that

$$y_0=\alpha_0p_0+\ \cdot\cdot\cdot\ +\alpha_{m-1}p_{m-1}.$$

Hence,

$$h=x_0+\alpha_0p_0+\ \cdot\cdot\cdot\ +\alpha_{m-1}p_{m-1}.$$

Moreover,

$$r_0=k-Ax_0=A(h-x_0)=\alpha_0Ap_0+\ \cdot\cdot\cdot+\alpha_{m-1}Ap_{m-1}.$$

[1] This method was presented from a different point of view by Fox, Huskey, and Wilkinson on p. 149 of a paper entitled "Notes on the solution of algebraic linear simultaneous equations," Quarterly Journal of Mechanics and Applied Mathematics c. 2, 149–173 (1948).

Computing the inner product (p_i,r_0) we find by (4:3a) and (4:4) that $\alpha_i=a_i$, and hence that $h=x_m$, as was to be proved.

The cd-method can be looked upon as a relaxation method. In order to establish this result, we introduce the function

$$f(x)=(h-x,A(h-x))=(x,Ax)-2(x,k)+(h,k). \qquad (4:5)$$

Clearly, $f(x)\geqq 0$ and $f(x)=0$ if, and only if, $x=h$. The function $f(x)$ can be used as a measure of the "goodness" of x as an estimate of h. Since it plays an important role in our considerations, it will be referred to as the *error function*. If p is a direction vector, we have the useful relation

$$f(x+\alpha p)=f(x)-2\alpha(p,r)+\alpha^2(p,Ap), \qquad (4:6)$$

where $r=k-Ax=A(h-x)$, as one readily verifies by substitution. Considered as a function of α, the function $f(x+\alpha p)$ has a minimum value at $\alpha=a$, where

$$a=\frac{(p,r)}{(p,Ap)}. \qquad (4:7)$$

This minimum value differs from $f(x)$ by the quantity

$$f(x)-f(x+ap)=a^2(p,Ap)=\frac{(p,r)^2}{(p,Ap)}. \qquad (4:8)$$

Comparing (4:7) with (4:1a), we obtain the first two sentences of the following result:

Theorem 4:3. *The point x_i minimizes $f(x)$ on the line $x=x_{i-1}+\alpha p_{i-1}$. At the i-th step the error $f(x_{i-1})$ is relaxed by the amount*

$$f(x_{i-1})-f(x_i)=\frac{(p_{i-1},r_{i-1})^2}{(p_{i-1},Ap_{i-1})}. \qquad (4:9)$$

In fact, the point x_i minimizes $f(x)$ on the i-dimensional plane P_i of points

$$x=x_0+\alpha_0p_0+\ldots+\alpha_{i-1}p_{i-1}, \qquad (4:10)$$

where α_0, ..., α_{i-1} are parameters. This plane contains the points x_0, x_1, ..., x_i.

In view of this result the cd-method is a method of relaxation of the error function $f(x)$. An iteration of the routine may accordingly be referred to as a relaxation.

In order to prove the third sentence of the theorem observe that at the point (4:10)

$$f(x)=f(x_0)-\sum_{j=0}^{i-1}[2\alpha_j(p_j,r_0)-\alpha_j^2(p_j,Ap_j)].$$

At the minimum point we have

$$\alpha_j=\frac{(p_j,r_0)}{(p_j,Ap_j)},$$

and hence $\alpha_j=a_j$, by (4:4). The minimum point is accordingly the point x_i, as was to be proved.

Geometrically, the equation $f(x)=\text{const.}$ defines an ellipsoid of dimension $n-1$. The point at which $f(x)$ has a minimum is the center of the ellipsoid and is the solution of $Ax=k$. The i-dimensional plane P_i, described in the last theorem, cuts the ellipsoid $f(x)=f(x_0)$ in an ellipsoid E_i of dimension $i-1$, unless E_i is the point x_0 itself. (In the cg-method, E_i is never degenerate, unless $x_0=h$.) The point x_i is the center of E_i. Hence we have the corollary:

Corollary 1. The point x_i is the center of the $(i-1)$-dimensional ellipsoid in which the i-dimensional plane P_i intersects the $(n-1)$-dimensional ellipsoid $f(x)=f(x_0)$.

Although the function $f(x)$ is the fundamental error function which decreases at each step of the relaxation, one is unable to compute $f(x)$ without knowing the solution h we are seeking. In order to obtain an estimate of the magnitude of $f(x)$ we may use the following:

Theorem 4:4. *The error vector $y=h-x$, the residual $r=k-Ax$, and the error function $f(x)$ satisfy the relations*

$$\frac{|r|^2}{\mu(r)}\leqq f(x)\leqq\frac{|r|^2}{\mu(y)}, \qquad (4:11)$$

where $\mu(z)$ is the Rayleigh quotient

$$\mu(z)=\frac{(z,Az)}{|z|^2}. \qquad (4:12)$$

The Rayleigh quotient of the error vector y does not exceed that of the residual r, that is,

$$\mu(y)\leqq\mu(r). \qquad (4:13)$$

Moreover,

$$\frac{|r|}{\mu(r)}\leqq|y|\leqq\frac{|r|}{\mu(y)}. \qquad (4:14)$$

The proof of this result is based on the Schwarzian quotients

$$\frac{(z,Az)}{(z,z)}\leqq\frac{(Az,Az)}{(z,Az)}\leqq\frac{(Az,A^2z)}{(Az,Az)}. \qquad (4:15)$$

The first of these follows from the inequality of Schwarz

$$|(p,q)|^2\leqq(p,p)(q,q) \qquad (4:16)$$

by choosing $p=z$, $q=Az$. The second is obtained by selecting $p=Bz$, $q=B^3z$, where $B^2=A$.

In order to prove theorem 4:4 recall that if we set $y=h-x$, then

$$r=k-Ax=A(h-x)=Ay$$

$$f(x)=(y,Ay)$$

by (4:5). Using the inequalities (4:15) with $z=y$, we see that

$$\mu(y)=\frac{(y,Ay)}{(y,y)}\leqq\frac{(Ay,Ay)}{(y,Ay)}=\frac{|r|^2}{f(x)}\leqq\frac{(Ay,A^2y)}{(Ay,Ay)}$$

$$=\frac{(r,Ar)}{(r,r)}=\mu(r).$$

This yields (4:11) and (4:13). Using (4:16) with $p=y$ and $q=r$ we find that

$$f(x)=(y,Ay)=(y,r)\leqq |y|\ |r|.$$

Hence

$$f(x)=\mu(y)|y|^2\leqq |y|\ |r|,$$

so that the second inquality in (4:14) holds. The first inequality is obtained from the relations

$$\frac{|r|^2}{\mu(r)}\leqq f(x)\leqq |y|\ |r|.$$

As is to be expected, any cd-method has within its routine a determination of the inverse A^{-1} of A. We have, in fact, the following:

Theorem 4:5. *Let* $p_0, \ldots, p_{n-1}$ *be n mutually conjugate nonzero vectors and let* p_{ij} *be the j-th component of* p_i. *The element in the j-th row and k-th column of* A^{-1} *is given by the sum*

$$\sum_{i=0}^{n-1}\frac{p_{ij}p_{ik}}{(p_i,Ap_i)}.$$

This result follows from the formula

$$h=\sum_{i=0}^{n-1}\frac{(p_i k)}{(p_i,Ap_i)}\,p_i$$

for the solution h of $Ax=k$, obtained by selecting $x_0=0$.

We conclude this section with the following:

Theorem 4:6. *Let* π_i *be the* $(n-i)$*-dimensional plane through* x_i *conjugate to the vectors* $p_0, p_1, \ldots, p_{i-1}$. *The plane* π_i *contains the points* $x_i, x_{i+1}, \ldots$ *and intersects the* $(n-1)$*-dimensional ellipsoid* $f(x)=f(x_i)$ *in an ellipsoid* E_i' *of dimension* $(n-i-1)$. *The center of* E_i' *is the solution h of* $Ax=k$. *The point* x_{i+1} *is the midpoint of the chord* C_i *of* E_i' *through* x_i, *which is parallel to* p_i. *In the cg-method the chord* C_i *is normal to* E_i' *at* x_i *and hence is in the direction of the gradient of* $f(x)$ *at* x_i *in* π_i.

The last statement will be established at the end of section 6. The equations of the plane π_i is given by the system

$$(Ap_j,x-x_i)=0\qquad (j=0,1,\ldots,i-1).$$

Since $p_i,p_{i+1},\ldots$ are conjugate to $p_0,\ldots,p_{i-1}$, so also is

$$x_k-x_i=a_ip_i+\ldots+a_{k-1}p_{k-1}\qquad (k>i).$$

The points $x_i,x_{i+1},\ldots,x_m=h$ are accordingly in π_i, and h is the center of E_i'. The chord C_i is defined by the equation $x=x_i+ta_ip_i$, where t is a parameter. As is easily seen,

$$f(x_i+ta_ip_i)=f(x_i)-(2t-t^2)a_i^2(p_i,Ap_i).$$

The second endpoint of the chord C_i is the point $x_i+2a_ip_i$ at which $t=2$. The midpoint corresponds to $t=1$, and hence is the point x_{i+1} as was to be proved.

In view of theorem 4:6, it is seen that at each step of the cd-routine the dimensionality of the space π_i in which we seek the solution h is reduced by unity. Beginning with x_0, we select an arbitrary chord C_0 of $f(x)=f(x_0)$ through x_0 and find its center. The plane π_1 through x_1 conjugate to C_0 contains the centers of all chords parallel to C_0. In the next step we restrict ourselves to π_1 and select an arbitrary chord C_1 of $f(x)=f(x_1)$ through x_1 and find its midpoint x_2 and the plane π_2 in π_1 conjugate to C_1 (and hence to C_0). This process when repeated will yield the answer in at most n steps. In the cg-method the chord C_i of $f(x)=f(x_i)$ is chosen to be the normal at x_i.

5. Basic Relations in the cg-Method

Recall that in the cg-method the following formulas are used

$$p_0=r_0=k-Ax_0\tag{5:1a}$$

$$a_i=\frac{|r_i|^2}{(p_i,Ap_i)}\tag{5:1b}$$

$$x_{i+1}=x_i+a_ip_i\tag{5:1c}$$

$$r_{i+1}=r_i-a_iAp_i\tag{5:1d}$$

$$b_i=\frac{|r_{i+1}|^2}{|r_i|^2}\tag{5:1e}$$

$$p_{i+1}=r_{i+1}+b_ip_i.\tag{5:1f}$$

One should verify *that eq.* (5:1e) *and* (5:1f) *hold for* $i=0,1,2,\ldots$ *if, and only if,*

$$p_k=|r_k|^2\sum_{j=0}^{k}\frac{r_j}{|r_j|^2}\qquad k=0,1,2,\ldots\tag{5:2}$$

The present section is devoted to consequences of these formulas. As a first result, we have

Theorem 5:1. *The residuals* $r_0, r_1, \ldots$ *and the direction vectors* $p_0, p_1, \ldots$ *generated by* (5:1) *satisfy the relations*

$$(r_i,r_j)=0\qquad (i\neq j)\tag{5:3a}$$

$$(p_i,Ap_j)=0\qquad (i\neq j)\tag{5:3b}$$

$$(p_i,r_j)=0\quad (i<j),\qquad (p_i,r_j)=|r_i|^2\quad (i\geqq j)\tag{5:3c}$$

$$(r_i,Ap_i)=(p_i,Ap_i),\ (r_i,Ap_j)=0\ (i\neq j, i\neq j+1)\tag{5:3d}$$

The residuals $r_0, r_1, \ldots$ *are mutually orthogonal and the direction vectors* $p_0, p_1, \ldots$ *are mutually conjugate.*

The proof of this result will be made by induction. The vectors r_0, $p_0=r_0$ and r_1 satisfy these relations since

$$(r_0,r_1)=(p_0,r_1)=|r_0|^2-a_0(r_0,Ap_0)=0$$

by (5:1b). Suppose that (5:3) holds for the vectors $r_0, \ldots, r_k$ and $p_0, \ldots, p_{k-1}$. To show that p_k can be adjoined to this set it is necessary to show that

$$(r_i,p_k)=|r_k|^2 \qquad (i\leqq k) \qquad (5:4a)$$

$$(p_i,Ap_k)=0 \qquad (i<k) \qquad (5:4b)$$

$$(r_k,Ap_i)=(p_k,Ap_i) \qquad (i\leqq k, i\neq k-1) \qquad (5:4c)$$

Equation (5:4a) follows at once from (5:2) and (5:3a). To prove (5:4b) we use (5:1d) and find that

$$(r_{i+1},p_k)=(r_i,p_k)-a_i(Ap_i,p_k).$$

By (5:4a) this becomes

$$|r_k|^2=|r_k|^2-a_i(Ap_i,p_k) \qquad (i<k).$$

Since $a_i>0$, eq (5:4b) holds. In order to establish (5:4c), we use (5:1f) to obtain

$$(p_k,Ap_i)=(r_k,Ap_i)+b_{k-1}(p_{k-1},Ap_i)=(r_k,Ap_i) \qquad (i\neq k-1).$$

It follows that (5:4c) holds and hence that (5:3) holds for the vectors $r_0, r_1, \ldots, r_k$ and $p_0, p_1, \ldots, p_k$.

It remains to show that r_{k+1} can be adjoined to this set. This will be done by showing that

$$(r_i,r_{k+1})=0 \qquad (i\leqq k) \qquad (5:5a)$$

$$(Ap_i,r_{k+1})=0 \qquad (i<k) \qquad (5:5b)$$

$$(p_i,r_{k+1})=0 \qquad (i\leqq k). \qquad (5:5c)$$

By (5:1d) we have

$$(r_i,r_{k+1})=(r_i,r_k)-a_k(r_i,Ap_k).$$

If $i<k$, the terms on the right are zero and (5:5a) holds. If $i=k$, the right member is zero by (5:1b) and (5:3d). Using (5:1d) again we have with $i<k$

$$0=(r_{k+1},r_{i+1})=(r_{k+1},r_i)-a_i(r_{k+1},Ap_i)=-a_i(r_{k+1},Ap_i).$$

Hence (5:5b) holds. The equation (5:5c) follows from (5:5a) and the formula (5:2) for p_i.

As a consequence of (5:3b) we have the first two sentences of the following:

Theorem 5:2. The cg-method is a cd-method. It is the special case of the cd-method in which the p_i are obtained by A-orthogonalization of the residual vectors r_i. On the other hand, a cd-method in which the residuals $r_0, r_i, \ldots$ are mutually orthogonal is essentially a cg-method.

The term "essentially" is used to designate that we disregard iterations that terminate in fewer than n steps, unless one adds the natural assumption that the formula for p_i in the routine depends continuously on the initial estimate x_0. To prove this result we accordingly suppose that the routine terminates at the n-th step. Since the r_i is orthogonal to r_{i+1} we have $x_i\neq x_{i+1}$ and hence $a_i\neq 0$. It follows that $(p_i,r_i)\neq 0$ by (4:1a). We may accordingly suppose the vectors p_i have been normalized so that $(p_i,r_i)=|r_i|^2$. In view of (4:3b) and (4:3c) eq (5:3c) holds. Select numbers α_{ij} such that

$$p_i=\sum_{j=0}^{n-1}\alpha_{ij}r_j.$$

Taking the inner product of p_i with r_j it is seen by (5:3c) that

$$\alpha_{ij}=\frac{|r_i|^2}{|r_j|^2}\,(i\geqq j), \quad \alpha_{ij}=0 \qquad (<j).$$

Consequently, (5:2) holds and the theorem is established.

Theorem 5:3. The residual vectors $r_0, r_1, \ldots$ and the direction vectors $p_0, p_1, \ldots$ satisfy the further relations

$$(p_i,p_j)=\frac{|r_j|^2|p_i|^2}{|r_i|^2} \qquad (i\leqq j) \qquad (5:6a)$$

$$|p_i|^2=|r_i|^2+b_{i-1}^2|p_{i-1}|^2=|r_i|^4\sum_{j=0}^{i}\frac{1}{|r_j|^2} \qquad (i>0) \qquad (5:6b)$$

$$(r_i,Ar_j)=0 \qquad |i-j|>1 \qquad (5:6c)$$

$$(r_i,Ar_i)=(p_i,Ap_i)+b_{i-1}^2(p_{i-1},Ap_{i-1}) \qquad (i>0). \qquad (5:6d)$$

The vector r_i is shorter than p_i. The vector p_i makes an acute angle with p_j.

The relations (5:6a) and (5:6b) follow readily from (5:1e), (5:1f), (5:2), and (5:3). Using (5:1f) and (5:3d), we see that

$$(r_i,Ar_j)=(r_i,Ap_j)-b_{j-1}(r_i,Ap_{j-1})=0 \qquad (i<j-1).$$

Hence (5:6c) holds. Equation (5:6d) is a consequence of (5:1f) and (5:3b). The final statements are interpretations of formula (5:6b) and (5:6a).

Theorem 5:4. The direction vectors $p_0, p_1, \ldots$ satisfy the relations

$$p_1=(1+b_0)p_0-a_0Ap_0 \qquad (5:7a)$$

$$p_{i+1}=(1+b_i)p_i-a_iAp_i-b_{i-1}p_{i-1} \qquad (i>0). \qquad (5:7b)$$

Similarly, the residuals $r_0, r_1, \ldots$ satisfy the relations

$$r_1=r_0-a_0Ar_0 \qquad (5:8a)$$

$$r_{i+1}=(1+b'_{i-1})r_i-a_iAr_i-b'_{i-1}r_{i-1}, \qquad (5:8b)$$

where

$$b_{i-1}=\frac{a_i}{a_{i-1}}\,b_{i-1}. \qquad (5:9)$$

Equation (5:7b) is obtained by eliminating r_{i+1} and r_i from the equations

$$p_{i+1}=r_{i+1}+b_ip_i$$

$$r_{i+1}=r_i-a_iAp_i$$

$$p_i=r_i+b_{i-1}p_{i-1}.$$

Equation (5:7a) follows similarly. In order to prove (5:8b), eliminate Ap_i and Ap_{i-1} from the equations

$$r_{i+1}=r_i-a_iAp_i$$

$$Ap_i=Ar_i+b_{i-1}Ap_{i-1}$$

$$r_i=r_{i-1}-a_{i-1}Ap_{i-1}.$$

Equation (5:8a) holds since $p_0=r_0$.

Theorem 5:5. *The scalars a_i and b_i are given by the several formulas*

$$a_i=\frac{|r_i|^2}{(p_i,Ap_i)}=\frac{(p_i,r_i)}{(p_i,Ap_i)}=\frac{(p_i,r_0)}{(p_i,Ap_i)} \qquad (5:10)$$

$$b_i=\frac{|r_{i+1}|^2}{|r_i|^2}=-\frac{(r_{i+1},Ap_i)}{(p_i,Ap_i)}=-\frac{(r_{i+1},Ar_i)}{(p_i,Ap_i)}. \qquad (5:11)$$

The scalar a_i satisfies the relation

$$\frac{1}{a_0}=\mu(r_0), \qquad \mu(p_i)<\frac{1}{a_i}<\mu(r_i) \quad (i>0), \qquad (5:12)$$

where $\mu(z)$ is the Rayleigh quotient (4:12). *The reciprocal of a_i lies between the smallest and larjest characteristic roots of A.*

The formula (5:10) follows from (5:1b) and (5:3c), while (5:11) follows from (5:1e), (5:1f), (5:3b), and (5:3d). Since

$$|r_i|<|p_i|, \qquad (r_i,Ar_i)>(p_i,Ap_i)$$

by (5:6b) and (5:6d), we have

$$\frac{(p_i,Ap_i)}{|p_i|^2}<\frac{(p_i,Ap_i)}{|r_i|^2}<\frac{(r_i,Ar_i)}{|r_i|^2}.$$

The inequalities (5:12) accordingly hold. The last statement is immediate, since $\mu(z)$ lies between the smallest and largest characteristic roots of A.

6. Properties of the Estimates x_i of h in the cg-Method

Let now $x_0, x_1, \ldots, x_m=h$ be the estimates of h obtained by applying the cg-method. Let $r_0, r_1, \ldots, r_m=0$ be the corresponding residuals and $p_0, p_1, \ldots, p_{m-1}$ the direction vectors used. The present section will be devoted to the study of the properties of the points $x_0, x_1, \ldots, x_m$. As a first result we have

Theorem 6:1. *The estimates $x_0, x_1, \cdots, x_m$ of h are distinct. The point x_i minimizes the error function $f(x)=(h-x,A(h-x))$ on the i-dimensional plane P_i passing through the points $x_0, x_1, \cdots, x_i$. In the* ith *step of the cg-method, $f(x)$ is diminished by the amount*

$$f(x_{i-1})-f(x_i)=a_{i-1}|r_{i-1}|^2=\mu(p_{i-1})|x_{i-1}-x_i|^2, \qquad (6:1)$$

where $\mu(z)$ is the Rayleigh quotient (4:12). *Hence,*

$$f(x_i)-f(x_j)=a_i|r_i|^2+\cdots+a_{j-1}|r_{j-1}|^2 \qquad (i<j). \qquad (6:2)$$

The point x_i is given by the formulas

$$x_i=x_0+\sum_{j=0}^{i-1}a_jp_j=x_0+\sum_{j=0}^{i-1}\frac{f(x_j)-f(x_i)}{|r_j|^2}r_j. \qquad (6:3)$$

This result is essentially a restatement of theorem 4:3. The formula (6:3) follows from (5:2) and (6:2). The second equation in (6:1) is readily verified.

Theorem 6:2. *Let S_i be the convex closure of the estimates $x_0, x_1, \cdots, x_i$. The point x_i is the point in S_i, whose error vector $h-x$ is the shortest.*

For a point $x\neq x_i$ in S_i is expressible in the form

$$x=\alpha_0x_0+\cdots+\alpha_ix_i,$$

where $\alpha_l\geqq 0$, $\alpha_0+\alpha_1+\cdots+\alpha_i=1$.

We have accordingly

$$x_i-x=\alpha_0(x_i-x_0)+\cdots+\alpha_{i-1}(x_i-x_{i-1})=\beta_0p_0+\cdots+\beta_{i-1}p_{i-1},$$

where the β's are nonnegative. Inasmuch as all $(p_j,p_k)>0$ it follows that

$$(x_j-x_i, x_i-x)>0 \qquad (i<j).$$

Using the relation

$$|x_j-x|^2=|x_j-x_i|^2+2(x_j-x_i, x_i-x)+|x_i-x|^2 \qquad (6:4)$$

we find that

$$|x_j-x_i|<|x_j-x| \qquad (i<j).$$

Setting $j=m$, we obtain theorem 6:2.

Incidentally, we have established the

Corollary. The point x_i is the point in S_i nearest to the point x_j $(j>i)$.

Theorem 6:3. *At each step of the cg-algorithm the error vector $y_i=h-x_i$ is reduced in length. In fact,*

$$|y_{i-1}|^2-|y_i|^2=\frac{f(x_i)+f(x_{i-1})}{\mu(p_{i-1})}, \qquad (6:5)$$

where $\mu(z)$ is the Rayleigh quotient (4:12).

In order to establish (6:5) observe that, by (5:6a),

$$\begin{aligned}(y_i.x_i-x_{i-1})&=(x_m-x_i,p_{i-1})a_{i-1}\\ &=[a_i(p_i,p_{i-1})+\ldots+a_{m-1}(p_{m-1},p_{i-1})]a_{i-1}\\ &=[a_i|r_i|^2+\ldots+a_{m-1}|r_{m-1}|^2]\frac{a_{i-1}|p_{i-1}|^2}{|r_{i-1}|^2}.\end{aligned}$$

In view of (6:2) and (5:1b) this becomes

$$(y_i,x_i-x_{i-1})=\frac{f(x_i)}{\mu(p_{i-1})}. \qquad (6:6)$$

Setting $x=x_{i-1}$ and $j=m$ in (6:4), we obtain (6:5) by the use of (6:6) and (6:1).

This result establishes the cg-method as a method of successive approximations and justifies the procedure of stopping the algorithm before the final-step is reached. If this is done, the estimate obtained can be improved by using the results given in the next two theorems.

Theorem 6:4. *Let $x_{i+1}^{(i)}, \cdots, x_m^{(i)}$ be the projections of the points $x_{i+1}, \cdots, x_m=h$ in the i-dimensional plane P_i passing through the points x_0, $x_1, \cdots, x_i$. The points $x_{i-1}, x_i, x_{i+1}^{(i)}, \cdots, x_m^{(i)}$ lie on a straight line in the order given by their enumeration. The point $x_k^{(i)}$ $(k>i)$ is given by the formulas*

$$x_k^{(i)}=x_{i-1}+\frac{f(x_{i-1})-f(x_k)}{f(x_{i-1})-f(x_i)}(x_i-x_{i-1}), \qquad (6:7a)$$

$$x_k^{(i)}=x_{i-1}+\frac{f(x_i)-f(x_k)}{|r_{i-1}|^2}p_{i-1}. \qquad (6:7b)$$

In order to prove this result, it is sufficient to establish (6:7). To this end observe first that the vector

$$p_j-\frac{|r_j|^2}{|r_{i-1}|^2}p_{i-1} \qquad (j\geqq i)$$

is orthogonal to each of the vectors $p_0, p_1, \cdots, p_{i-1}$. This can be seen by forming the inner product with $p_l(l<i)$, and using (5:6a). The result is

$$(p_l,p_j)-\frac{|r_j|^2}{|r_{i-1}|^2}(p_l,p_{i-1})=\frac{|p_l|^2}{|r_l|^2}[|r_j|^2-|r_j|^2]=0$$

The projection of the point

$$x_k=x_{i-1}+a_{i-1}p_{i-1}+a_ip_i+\ldots+a_{k-1}p_{k-1}$$

in P_i is accordingly

$$x_k^{(i)}=x_{i-1}+\frac{a_{i-1}|r_{i-1}|^2+\ldots+a_{k-1}|r_{k-1}|^2}{|r_{i-1}|^2}p_{i-1}.$$

Using (6:2), we obtain (6:7). The points lie in the designated order, since $f(x_k)>f(x_{k+1})$.

Since $f(x_m)=0$, we have the first part of

Theorem 6:5. *The point*

$$x_m^{(i)}=x_i+\frac{f(x_i)}{|r_{i-1}|^2}p_{i-1} \qquad (6:8)$$

is the point in P_i whose distance from the solution h is the least. It lies on the line $x_{i-1}x_i$ beyond x_i. Moreover,

$$\frac{1}{f(x_m^{(i)})}=\frac{1}{f(x_i)}-\frac{1}{f(x_{i-1})} \qquad (6.9)$$

and

$$|h-x_i|^2=|h-x_m^{(i)}|^2+\frac{f(x_m^{(i)})-f(x_i)}{\mu(p_{i-1})}. \qquad (6.10)$$

In order to establish (6:9) and 6:10) we use the formula

$$f(x_i+\alpha p_{i-1})=f(x_i)+\alpha^2(p_{i-1},Ap_{i-1}),$$

which holds for all values of α in view of the fact that x_i minimizes $f(x)$ on P_i. Setting $\alpha=f(x_i)/|r_{i-1}|^2$ we have

$$f(x_m^{(i)})=f(x_1)+\frac{f(x_i)^2}{a_{i-1}|r_{i-1}|^2}=f(x_i)^2+\frac{f(x_i)^2}{f(x_{i-1})-f(x_i)}.$$

An algebraic reduction yields (6:9). Inasmuch as

$$\begin{aligned}|h-x_i|^2-|h-x_m^{(i)}|^2&=\frac{f(x_i)^2|p_{i-1}|^2}{|r_{i-1}|^4}\\ &=\frac{f(x_i)^2}{f(x_{i-1})-f(x_i)}\frac{a_{i-1}|p_{i-1}|^2}{|r_{i-1}|^2},\end{aligned}$$

we obtain (6:10) from (6:9) and (5:1b).

As a further result we have

Theorem 6:6. *Let $x_1', \ldots, x_{m-1}'$ be the projections of the points $x_1, \ldots, x_{m-1}'$ on the line joining the initial point x_0 to the solution $x_m=h$. The points $x_0, x_1', \ldots, x_{m-1}', x_m=h$ lie in the order of enumeration.*

Thus, it is seen that we proceed towards the solution without oscillation. To prove this fact we need only observe that

$$\begin{aligned}(x_m-x_0,x_i-x_{i-1})&=(x_m-x_0,a_{i-1},p_{i-1})\\ &=a_{i-1}\sum_{j=0}^{m-1}a_j(p_j,p_{i-1})>0\end{aligned}$$

by (5:6a). A similar result holds for the line joining x_i to $x_j(i<j)$.

Let π_i be the $(n-i)$-dimensional plane through x_i conjugate to $p_0, p_1, \ldots, p_{i-1}$. It consists of the set of points x satisfying the equation

$$(Ap_j,x-x_i)=0 \qquad (j=0,1,\ldots,i-1).$$

This plane contains the points $x_{i+1}, \ldots, x_m$ and hence the solution h.

Theorem 6:7. *The gradient of the function $f(x)$ at x_i in the plane π_i is a scalar multiple of the vector p_i.*

The gradient of $f(x)$ at x_i is the vector $-r_i$. The gradient q_i of $f(x)$ at x_i in π_i is the orthogonal projection of $-r_i$ in the plane π_i. Hence q_i is of the form

$$q_i=-r_i+\alpha_0Ap_0+\ldots+\alpha_{i-1}Ap_{i-1},$$

where $\alpha_0, \ldots, \alpha_{i-1}$ are chosen so that q_i is orthogonal to $Ap_0, \ldots, Ap_{i-1}$. Since

$$p_i = |r_i|^2 \sum_{j=0}^{i} \frac{r_j}{|r_j|^2}$$

$$r_{j+1} = r_j - a_j A p_j \qquad (j=0,1, \ldots, i-1),$$

it is seen upon elimination of $r_0, r_1, \ldots, r_{i-1}$ successively that p_i is also a linear combination of $r_i, Ap_0, \ldots, Ap_{i-1}$. Inasmuch as p_i is conjugate to $p_0, \ldots, p_{i-1}$, it is orthogonal to $Ap_0, \ldots, Ap_{i-1}$. The vector p_i accordingly is a scalar multiple of the gradient q_i of $f(x)$ at x_i in π_i, as was to be proved.

In view of the result obtained in theorem 6:7 it is seen that the name "method of conjugate gradients" is an appropriate name for the method given in section 3. In the first step the relaxation is made in the direction p_0 of the gradient of $f(x)$ at x_0, obtaining a minimum value of $f(x)$ at x_1. Since the solution h lies in π_1, it is sufficient to restrict x to the plane π_1. Accordingly, in the next step, we relax in the direction p_1 of the gradient of $f(x)$ in π_1 at x_1, obtaining the point x_2 at which $f(x)$ is least. The problem is then reduced to relaxing $f(x)$ in the plane π_2, conjugate to p_0 and p_1. At the next step the gradient in x_2 in π_2 is used, and so on. The dimensionality of the space in which the relaxation is to take place is reduced by unity at each step. Accordingly, after at most n steps, the desired solution is attained.

7. Properties of the Estimates $\bar{x}_i$ of h in the cg-Method

In the cg-method there is a second set of estimates $\bar{x}_0 = x_0, \bar{x}_1, \bar{x}_2, \ldots$ of h that can be computed, and that are of significance in application to linear systems arising from difference equations approximating boundary-valve problems. In these applications, the function defined by $\bar{x}_i$ is smoother than that of x_i, and from this point of view is a better approximation of the solution h. The point $\bar{x}_i$ has its residual proportional to the conjugate gradient p_i. The points $\bar{x}_0, \bar{x}_1, \bar{x}_2, \ldots$ can be computed by the iteration (7:2) given in the following:

Theorem 7:1. *The conjugate gradient p_i is expressible in the form*

$$p_i = c_i(k - A\bar{x}_i), \tag{7:1}$$

where c_i and $\bar{x}_i$ are defined by the recursion formulas

$$c_0 = 1, \; c_{i+1} = 1 + b_i c_i \tag{7:2a}$$

$$\bar{x}_0 = x_0, \; \bar{x}_{i+1} = \frac{x_{i+1} + b_i c_i \bar{x}_i}{c_{i+1}}. \tag{7:2b}$$

We have the relations

$$c_i = |r_i|^2 \sum_{j=0}^{i} \frac{1}{|r_j|^2} = \frac{|p_i|^2}{|r_i|^2} \tag{7:3a}$$

$$\bar{x}_i = \frac{|r_i|^2}{c_i} \sum_{j=0}^{i} \frac{x_j}{|r_j|^2} \tag{7:3b}$$

$$\bar{r}_i = k - A\bar{x}_i = \frac{1}{c_i} p_i = \frac{|r_i|^2}{c_i} \sum_{j=0}^{i} \frac{r_j}{|r_j|^2}. \tag{7:3c}$$

The sum of the coefficients of $x_0, x_1, \ldots, x_i$ in (7:3b) (and hence of $r_0, r_1, \ldots, r_i$ in (7:3c)) is unity.

The relation (7:1) can be established by induction. It holds for $i=0$. If it holds for i, then

$$\begin{aligned} p_{i+1} = r_{i+1} + b_i p_i = (1 + b_i c_i)k - A(x_{i+1} + b_i c_i \bar{x}_i) \\ = c_{i+1}(k - A\bar{x}_{i+1}). \end{aligned}$$

The formula (7:3a) follows from (7:2a), (5:1e) and (5:6b). Formula (7:3b) is an easy consequence of (7:2b). To prove (7:3c) one can use (5:2) or (7:3b), as one wishes. The final statement is a consequence of (7:3a).

Theorem 7:2. *The point $\bar{x}_i$ given by* (7:2) *lies in the convex closure S_i of the points $x_0, x_1, \cdots, x_i$. It is the point x in the i-dimensional plane P_i through $x_0, x_1, \cdots, x_i$ at which the squared residual $|k - Ax|^2$ has its minimum value. This minimum value is given by the formula*

$$|\bar{r}_i|^2 = \frac{|r_i|^2}{c_i} = \frac{|r_i|^4}{|p_i|^2}. \tag{7:4}$$

The squared residuals $|r_0|^2, |\bar{r}_i|^2, \cdots$ diminish monotonically during the cg-method. At the ith step the squared residual is reduced by the amount

$$|\bar{r}_{i-1}|^2 - |\bar{r}_i|^2 = \frac{|\bar{r}_{i-1}|^2}{c_i}. \tag{7:5}$$

The first statement follows from (7:3b), since the coefficients of $x_0, x_1, \cdots, x_i$ are positive and have unity as their sum. In order to show that the squared residual has a minimum on P_i at $\bar{x}_i$, observe that a point x in P_i differs from $\bar{x}_i$ by a vector z_i of the form

$$x - \bar{x}_i = z_i = \alpha_0 p_0 + \cdots + \alpha_{i-1} p_{i-1}.$$

The residual $r = k - Ax$ is accordingly given by

$$r = \bar{r}_i - Az_i$$

$$Az_i = \alpha_0 A p_0 + \cdots - \alpha_{i-1} A p_{i-1}.$$

Inasmuch as, by (7:3c), $\bar{r}_i = p_i / c_i$, we have

$$(\bar{r}_i, Ap_j) = \frac{1}{c_i}(p_i, Ap_j) = 0 \qquad (j < i).$$

Consequently, $(\bar{r}_i, Az_i) = 0$ and

$$|r|^2 = |\bar{r}_i|^2 + |Az_i|^2 > |\bar{r}_i|^2 \qquad (x \neq \bar{x}_i).$$

It follows that $\bar{x}_i$ affords a proper minimum to $|r|^2$ on P_i. Using (7:3c) and (7:3a) and the orthogonality of r_j's, it is seen that the minimum value of $|r|^2$ on P_i is given by (7:4). By (7:4) and (7:2a)

$$|\bar{r}_{i-1}|^2-|\bar{r}_i|^2=|\bar{r}_{i-1}|^2\left(1-\frac{c_{i-1}}{c_i}\frac{|r_i|^2}{|r_{i-1}|^2}\right)$$

$$=|\bar{r}_{i-1}|^2\left(1-\frac{b_{i-1}c_{i-1}}{c_i}\right)=\frac{|\bar{r}_{i-1}|^2}{c_i}.$$

This completes the proof of theorem 7:1.

Theorem 7:3. The Rayleigh quotients of r_0, r_1, . . . and $\bar{r}_0$, $\bar{r}_1$, . . . are connected by the formulas

$$\frac{\mu(r_i)}{|r_i|^2}=\frac{\mu(\bar{r}_i)}{|\bar{r}_i|^2}+\frac{\mu(\bar{r}_{i-1})}{|\bar{r}_{i-1}|^2} \tag{7:6a}$$

$$\frac{\mu(\bar{r}_i)}{|\bar{r}_i|^2}=\frac{\mu(r_i)}{|r_i|^2}-\frac{\mu(r_{i-1})}{|r_{i-1}|^2}+\cdot\cdot\cdot+(-1)^i\frac{\mu(r_0)}{|r_0|^2}. \tag{7:6b}$$

The Rayleigh quotient of $\bar{r}_i(i>0)$ is smaller than that of r_i, that is, $\mu(\bar{r}_i)<\mu(r_i)$.

In order to prove this result we use (5:6d) and obtain

$$\frac{(r_i,Ar_i)}{|r_i|^4}=\frac{(p_i,Ap_i)}{|r_i|^4}+\frac{(p_{i-1},Ap_{i-1})}{|r_{i-1}|^4}.$$

Since $|r_i|^4=|p_i|^2|\bar{r}_i|^2$ and $\mu(p_i)=\mu(\bar{r}_i)$, this relation yields (7:6a). The eq (7:6b) follows from (7:6a). The last statement follows from (5:12).

In the applications to linear systems arising from difference equations approximating boundary value problems, $\mu(r_i)$ can be taken as a measure of the smoothness of x_i. The smaller $\mu(r_i)$ is, the smoother x_i is. *Hence $\bar{x}_i$ is smoother than x_i.*

Theorem 7:4. At the point $\bar{x}_i$ the error function $f(x)$ has the value

$$f(\bar{x}_i)=f(x_i)+|\bar{r}_i|^4\sum_{j=0}^{i-1}\frac{f(x_j)-f(x_{j+1})}{|\bar{r}_j|^4} \tag{7:7}$$

and we have

$$f(x_i)<f(\bar{x}_i)<f(\bar{x}_{i-1}). \tag{7:8}$$

The sequence $f(\bar{x}_0)$, $f(\bar{x}_1)$, $f(\bar{x}_2)$, . . . decreases monotonically.

In order to prove this result, it is convenient to set

$$\bar{b}_{i-1}=\frac{|\bar{r}_i|^2}{|\bar{r}_{i-1}|^2}=\frac{c_{i-1}|r_i|^2}{c_i|r_{i-1}|^2}=\frac{b_{i-1}c_{i-1}}{c_i}. \tag{7:9}$$

By (7:2) we have the relation

$$\bar{x}_i-x_i=\bar{b}_{i-1}(\bar{x}_{i-1}-x_i). \tag{7:10}$$

Using the formula

$$f(x)-f(x_i)=(x-x_i,A(x-x_i)),$$

which holds for any x in P_i, we see that

$$f(\bar{x}_i)-f(x_i)=\bar{b}^2_{i-1}(\bar{x}_{i-1}-x_i,A(\bar{x}_{i-1}-x_i)),$$

that is,

$$f(\bar{x}_i)-f(x_i)=\bar{b}^2_{i-1}[f(\bar{x}_{i-1})-f(x_i)]. \tag{7:11}$$

By (7:9) it is seen that this result can be put in the form

$$\frac{f(\bar{x}_i)-f(x_i)}{|\bar{r}_i|^4}=\frac{f(x_{i-1})-f(x_i)}{|\bar{r}_{i-1}|^4}+\frac{f(\bar{x}_{i-1})-f(x_{i-1})}{|\bar{r}_{i-1}|^4}.$$

Since $\bar{x}_0=x_0$ this formula yields the desired relation (7:7). Since $\bar{b}_{i-1}<1$, it follows from (7:11) and (7:7) that (7:8) holds. This proves the theorem.

Theorem 7:5. The error vector $y_i=h-x_i$ is shorter than the error vector $\bar{y}_i=h-\bar{x}_i$. Moreover, $\bar{y}_i$ is shorter than y_{i-1}.

The first statement follows from (7:2). It also follows from theorem 6:2, since $\bar{x}_i$ is in S_i. By (7:2) the point $\bar{x}_i$ lies in the line segment joining x_i to $\bar{x}_{i-1}$. The distance from h to $\bar{x}_i$ exceeds the distance from h to x_i. It follows that as we move from x_i to $\bar{x}_{i-1}$ the distance from h is increased, as was to be proved.

8. Propagation of Rounding-Off Errors in the cg-Method

In this section we take as basic relations between the vectors r_0, r_1, $\cdot\cdot\cdot$ and p_0, p_1, $\cdot\cdot\cdot$ in the cg-method the following:

$$p_0=r_0, \tag{8:1a}$$

$$a_i=\frac{|r_i|^2}{(p_i,Ap_i)} \tag{8:1b}$$

$$r_{i+1}=r_i-a_iAp_i \tag{8:1c}$$

$$b_i=\frac{|r_{i+1}|^2}{|r_i|^2} \tag{8:1d}$$

$$p_{i+1}=r_{i+1}+b_ip_i. \tag{8:1e}$$

As a consequence, we have the orthogonality relations

$$(r_i,r_k)=0,\quad (Ap_i,p_k)=0\qquad (i\neq k). \tag{8:2}$$

Because of rounding-off errors during a numerical calculation (routine), these relations will not be satisfied exactly. As the difference $|k-i|$ increases, the error in (8:2) may increase so rapidly that x_n will not be as good an estimate of h as desired. This error can be lessened in two ways: first, by introducing a subsidiary calculation to reduce rounding-off errors; and second, by repeating the iteration so as to obtain a new estimate. This section will be concerned with the first of these and with a study of the propagation of the rounding-off errors. To this end it is convenient to divide the section in four parts, the first of which is the following:

8.1. Basic propagation formulas

In this part we derive simple formulas showing how errors in scalar products of the type

$$(r_{i-1},r_i), \qquad (Ap_{i-1},p_i) \qquad (8:3)$$

are propagated during the next step of the computation. From (8:1e) follows

$$(r_i,r_{i+1})=(p_i,r_{i+1})-b_{i-1}(p_{i-1},r_{i+1}).$$

Inserting (8:1c) in both terms on the right yields

$$(r_i,r_{i+1})=(p_i,r_i)-a_i(p_i,Ap_i)-b_{i-1}(p_{i-1},r_i)+b_{i-1}a_i(Ap_{i-1},p_i).$$

Applying (8:1e) to the first and third terms gives

$$(r_i,r_{i+1})=(r_i,r_i)-a_i(p_i,Ap_i)+b_{i-1}a_i(Ap_{i-1},p_i), \quad (8:4)$$

which by (8:1b) becomes

$$(r_i,r_{i+1})=b_{i-1}a_i(Ap_{i-1},p_i). \qquad (8:5)$$

This is our first propagation formula.

Using (8:le) again,

$$(Ap_i,p_{i+1})=Ap_i,r_{i+1})+b_i(Ap_i,p_i).$$

Inserting (7:1c) in the first term,

$$(Ap_i,p_{i+1})=-\frac{1}{a_i}|r_{i+1}|^2+\frac{1}{a_i}(r_i,r_{i+1})+b_i(Ap_i,p_i). \qquad (8:6)$$

But in view of (8:1b) and (8:1d)

$$|r_{i+1}|^2=a_ib_i(Ap_i,p_i). \qquad (8:7)$$

Therefore,

$$(Ap_i,p_{i+1})=\frac{1}{a_i}(r_i,r_{i+1}). \qquad (8:8)$$

This is our *second propagation formula.*

Putting (8:5) and (8:8) together yields the *third and fourth propagation formulas*

$$(r_i,r_{i+1})=\frac{b_{i+1}a_i}{a_{i-1}}(r_{i-1},r_i) \qquad (8:9a)$$

$$(Ap_i,p_{i+1})=b_{i-1}(Ap_{i-1},p_i), \qquad (8:9b)$$

which can be written in the alternate form

$$\frac{(r_i,r_{i+1})}{|r_i|^2}=\frac{a_i}{a_{i-1}}\frac{(r_{i-1},r_i)}{|r_{i-1}|^2} \qquad (8:10a)$$

$$\frac{(Ap_i,p_{i+1})}{(Ap_i,p_i)}=\frac{a_i}{a_{i-1}}\frac{(Ap_{i-1},p_i)}{(Ap_{i-1},p_{i-1})} \qquad (8:10b)$$

by virtue of (8:1b) and (8:1d). Each of these propagation formulas, and in particular the simple formulas (8:9), can be used to check whether nonvanishing products (8:3) are due to normal rounding-off errors or to errors of the computer. The formulas (8:10) have the following meaning. If we build the symmetric matrix P having the elements (Ap_i,p_k), the left side of (8:10b) is the ratio of two consecutive elements in the same line, one located in the main diagonal and one on its right hand side. The formula (8:10b) gives the change of this ratio as we go down the main diagonal.

8.2. A Stability Condition

Even if the scalar products (8:2) are not all zero, so that the vectors $p_0, p_1, \cdots, p_{n-1}$ are not exactly conjugate, we may use these vectors for solving $Ax=k$ in the following way. The solution h may be written in the form

$$h=x_0+a_0'p_0+a_1'p_1+\cdots+a_{n-1}'p_{n-1}. \qquad (8:11)$$

Taking the scalar product with Ap_i, we obtain

$$(x_0,Ap_i)+\sum_k(Ap_i,p_k)a_k'=(h,Ap_i)=(Ah,p_i)=(k,p_i)$$

or

$$\sum_k(Ap_i,p_k)a_k'=(r_0,p_i). \qquad (8:12)$$

The system $Ax=k$ may be replaced by this linear system for $a_0',\cdots,a_{n-1}'$. Therefore, because of rounding-off errors we have certainly not solved the given system exactly, but we have reached a more modest goal, namely, we have transformed the given system into the system (8:12), which has a dominating main diagonal if rounding-off errors have not accumulated too fast. The cg-algorithm gives an approximate solution

$$h'=x_0+a_0p_0+\cdots+a_{n-1}p_{n-1}. \qquad (8:13)$$

A comparison of (8:11) and (8:13) shows that the number a_k computed during the cg-process is an approximate value of a_k'.

In order to have a dominating main diagonal in the matrix of the system (8:12) the quotients

$$\frac{(Ap_i,p_k)}{(Ap_i,p_i)} \qquad (i\neq k) \qquad (8:14)$$

must be small. In particular this must be true for $k=i+1$. In this special case we learn from (8:10b) that increasing numbers $a_0, a_1,\cdots$ during the cg-process lead to accumulation of rounding-off errors, because then these quotients increase also. We have accordingly the following stability condition.

The larger the ratios a_i/a_{i-1}, the more rapidly the rounding-off errors accumulate.

A more elaborate discussion of the general quotient (8:14) gives essentially the same result.

By theorem 5:5, the scalars a_i lie on the range

$$\frac{1}{\lambda_{\max}}<a_i<\frac{1}{\lambda_{\min}},$$

where $\lambda_{\min}$, $\lambda_{\max}$ are the least and the largest eigenvalues of A. Accordingly, the ratio $\rho=\lambda_{\max}/\lambda_{\min}$ is an upper bound of the critical ratio a_i/a_{i-1}, which determines the stability of the process. When ρ is near one, that is, when A is near a multiple of the identity, the cg-method is relatively stable. It will be shown in section 18 that examples can be constructed in which the ratios a_i/a_{i-1} $(i=1,\cdot\cdot\cdot,n-1)$ are any set of preassigned positive numbers. Thus the stability may be low. However, this instability can be compensated to a certain extent by starting the cg-process with a vector x_0 whose residual vector r_0 is near to the eigenvector of A corresponding to $\lambda_{\min}$. In this event a_0 is near to the upper bound $1/\lambda_{\min}$ of the a_i. This result is brought out in the following theorem:

For a given symmetric and positive definite matrix A, which has distinct eigenvalues, there exists always an initial residual vector r_0 such that $(a_i/a_{i-1})<1$ and hence such that the algorithm is stable with respect to the propagation of rounding-off errors.

In order to prove this we introduce the eigenvalues

$$\lambda_{\min}=\lambda_0<\lambda_1<\lambda_2<\ldots<\lambda_{n-1}=\lambda_{\max}$$

of A, and we take the corresponding normalized eigenvectors as a coordinate system. Let α_0, α_1, ..., α_{n-1} be real numbers not equal to zero and ϵ a small quantity. Then we start with a residual vector

$$r_0=(\alpha_0,\alpha_1\epsilon,\alpha_2\epsilon^2,\ldots,\alpha_{n-1}\epsilon^{n-1}). \qquad (8:14a)$$

Expanding everything in a power series, one finds that

$$a_i=\frac{1}{\lambda_i}+\epsilon^2(*). \qquad (8:14b)$$

Hence

$$\frac{a_i}{a_{i-1}}=\frac{\lambda_{i-1}}{\lambda_i}+\epsilon^2(*)<1$$

if ϵ is small enough.

As a by-product of such a choice of r_0 we get by (8:14b) approximations of the eigenvalues of A. Moreover, it turns out that in this case the successive residual-vectors r_0, r_1, ..., r_{n-1} are approximations of the eigenvectors.

These results suggest the following rule:

The cg-process should start with a smooth residual distribution, that is, one for which $\mu(r_0)$ is close to $\lambda_{\min}$. If needed, the first estimate can be smoothed by some relaxation process.

Of course, we may use for this preparing relaxation the cg-process itself, computing the estimates $\overline{x}_i$ given in section 7. A simpler method is to modify the cg-process by setting $b_i=0$ so that $p_i=r_i$ and selecting a_i of the form $a_i=\alpha/\mu(r_i)$, where α is a small constant in the range $0<\alpha<1$.

8.3. The End-Correction

The system (8:12) can be solved by ordinary relaxation processes. Introducing the numbers a_k as approximations of the solutions a'_k, we get the residuals

$$(r_0,p_i)-\sum_k (Ap_i,p_k)a_k. \qquad (8:15)$$

Inasmuch as $r_0=r_{i+1}+a_0Ap_0+\ldots+a_iAp_i$ by (8:1c), we have

$$(r_0,p_i)=(r_{i+1},p_i)+a_0(Ap_i,p_0)+\ldots+a_i(Ap_i,p_i). \qquad (8:16)$$

It follows that the residual (8:15) is reduced to

$$(r_{i+1},p_i)-(Ap_i,p_{i+1})a_{i+1}-(Ap_i,p_{i+2})a_{i+2}-\ldots$$
$$-(Ap_i,p_{n-1})a_{n-1}. \qquad (8:17)$$

This leads to the correction of a_i

$$\Delta a_i=\frac{1}{(Ap_i,p_i)}\{(r_{i+1},p_i)-(Ap_i,p_{i+1})a_{i+1}$$
$$-(Ap_i,p_{i+2})a_{i+2}-\ldots-(Ap_i,p_{n-1})a_{n-1}\}. \qquad (8:18)$$

A first approximation of a'_i is accordingly

$$a'_i\sim a_i+\Delta a_i.$$

In order to discuss this result, suppose that the numbers a_i have been computed accordingly to the formula

$$a_i=\frac{(p_i,r_i)}{(p_i,Ap_i)} \qquad (8:19)$$

(theorem 5:5). From (8:1c) it follows that $(r_{i+1},p_i)=0$. and therefore this term drops out in (8:18). In this case the correction Δa_i depends only on the products (Ap_i,p_k) with $i<k$. That is to say, that this correction is influenced only by the rounding-off errors *after* the i-th step. If, for instance, the rounding-off errors in the last 10 steps of a cg-process are small enough to be neglected, the last 10 values a_i need not to be corrected. Hence, generally, the Δa_i decrease rather rapidly.

From (8:18) we learn that in order to have a good rounding-off behavior, it is not only necessary to keep the products (p_k,Ap_i) $(i\neq k)$ small, but also to satisfy $(r_{i+1},p_i)=0$ as well as possible. Therefore, it may be better to compute the a_i from the formulas (8:19) rather than from (8:1b). We see this immediately, if we compare (8:19) with (8:1b); by (8:19) and (8:1e) we have

$$a_i=\frac{1}{(p_i,Ap_i)}\{|r_i|^2+b_{i-1}(r_i,p_{i-1})\}.$$

For ill-conditioned matrices, where a_i and b_i may become considerably larger than 1, the omitting of the second summand may cause additional errors. For the same reason, it is at least as important in these cases to use formula (3:2b) rather than (8:1d) for determining b_i, since by (3:2b) and (8:1c)

$$b_i=\frac{1}{a_i(p_i,Ap_i)}\{|r_{i+1}|^2-(r_{i+1},r_i)\}.$$

Here the second summand is not directly made zero by any of the two sets of formulas for a_i and b$_i$. The only orthogonality relations, which are directly fulfilled in the scope of exactitude of the numerical computation by the choice of a_i and b_i, are the following:

$$(r_{i+1},p_i)=0, \quad (p_{i+1},Ap_i)=0.$$

Therefore, we have to represent (r_{i+1},r_i) in terms of these scalar products:

$$(r_{i+1},r_i)=(r_{i+1},p_i)-b_{i-1}(r_i,p_{i-1})+a_ib_{i-1}(p_i,Ap_{i-1}).$$

From this expression we see that for large b_i and a_i the second and third terms may cause considerable rounding-off errors, which affect also the relation $(p_{i+1},Ap_i)=0$, if we use formula (8:1d) for b_i. This is confirmed by our numerical experiments (section 19).

From a practical point of view, the following formula is more advantageous because it avoids the computation of all the products (Ap_i,p_k). From (8:1c) follows

$$r_n=r_{i+1}-a_{i+1}Ap_{i+1}-a_{i+2}Ap_{i+2}-\ .\ .\ .\ -a_{n-1}Ap_{n-1}$$

$$(r_n,p_i)=(r_{i+1},p_i)-a_{i+1}(Ap_i,p_{i+1})$$
$$-.\ .\ .\ -a_{n+1}(Ap_i,p_{n-1}).$$

and we have the result

$$\Delta a_i=\frac{(r_n,p_i)}{(Ap_i,p_i)}. \tag{8:20}$$

This formula gives corrections of the a_i if, because of rounding-off errors, the residual r_n is not small enough.

8.4. Refinement of the cg-algorithm

In order to diminish rounding-off errors in the orthogonality of the residuals r_i we refine our general routine (8:1). After the ith step in the routine we compute (Ap_{i-1},p_i), which should be small. Going then to the $(i+1)$st step we replace a_i by a slightly different quantity $\bar{a}_i$ chosen so that $(r_i,r_{i+1})=0$. In order to perform this, we may use (8:4), which now must be written

$$(r_i,r_{i+1})=(r_i,r_i)-\bar{a}_i(Ap_i,p_i)+b_{i-1}\bar{a}_i(Ap_{i-1},p_i)=0$$

yielding

$$\bar{a}_i=\frac{|r_i|^2}{(Ap_i,p_i)-b_{i-1}(Ap_{i-1},p_i)}.$$

Introducing the correction factor

$$d_i=1-b_{i-1}\frac{(Ap_{i-1},p_i)}{(Ap_i,p_i)} \tag{8:21}$$

and taking into account the old value (8:lb) of a_i, this can be written in the form

$$\bar{a}_i=\frac{a_i}{d_i}. \tag{8:22}$$

Continuing in the general routine of the $(i+1)$st step we replace b_i by a number $\bar{b}_i$ in such a way that $(Ap_i,p_{i+1})=0$. We use (8:6), which now must be written in the form

$$-|r_{i+1}|^2+(r_i,r_{i+1})+\bar{a}_i\bar{b}_i(Ap_i,p_i)=0.$$

The term (r_i,r_{i+1}) vanishes by virtue of our choice of $\bar{a}_i$. Using (8:7), we see that $a_ib_i=\bar{a}_i\bar{b}_i$ and from (8:22)

$$\bar{b}_i=b_id_i. \tag{8:23}$$

Since rounding-off errors occur again, this subroutine can be used in the same way to improve the results in the $(i+1)$th step.

The corrections just described can be incorporated automatically in the general routine by replacing the formulas (3:1) by the following refinement:

$$\begin{aligned}
&p_0=r_0=k-Ax_0, \qquad d_0=1\\
&a_i=\frac{|r_i|^2}{(p_i,Ap_i)}\,\frac{1}{d_i}\\
&x_{i+1}=x_i+a_ip_i\\
&r_{i+1}=r_i-a_iAp_i\\
&b_i=\frac{|r_{i+1}|^2}{|r_i|^2}\,d_i\\
&p_{i+1}=r_{i+1}+b_ip_i\\
&d_{i+1}=1-b_i\frac{(Ap_{i+1},p_i)}{(Ap_{i+1},p_{i+1})}.
\end{aligned} \tag{8:24}$$

Another quite obvious, but numerically more laborious method of refinement goes along the following lines. After finishing the ith step, compute a product of the type (Ap_i,p_k) with $k<i$. Then replace p_i by

$$\bar{p}_i=p_i-\frac{(Ap_i,p_k)}{(Ap_k,p_k)}\,p_k. \tag{8:25}$$

The vector p_i is exactly conjugate to p_k. This method may be used in place of (8:24) in case $k=i-1$. It has the disadvantage that a *vector* must be corrected and not just *numbers*, as in (8:24).

9. Modifications of the cg-Process

In the cg-method given by (3:1), the lengths of the vectors $p_0, p_1, \ldots$ are at our disposal. In order to preserve significant figures in computations, it is desirable to have all the p_i of about the same magnitude. In order to aim at this goal, we normalized the p's by the formulas

$$p_0=r_0, \quad p_i=r_i+b_{i-1}p_{i-1} \qquad (i>0).$$

However other normalizations can be made. In order to see how this normalization appears, we replace p_i by d_ip_i in eq (3:1), where d_i is a scalar factor. This factor d_i is not the same as that given in section 8 but plays a similar role. The result is

$$\begin{aligned}
&r_0=k-Ax_0, \qquad p_0=\frac{r_0}{d_0}\\
&x_{i+1}=x_i+a_ip_i\\
&r_{i+1}=r_i-a_iAp_i\\
&p_{i+1}=\frac{r_{i+1}+b_ip_i}{d_{i+1}}\\
&a_i=\frac{|r_i|^2}{(p_i,Ap_i)d_i}=\frac{(p_i,r_i)}{(p_i,Ap_i)}\\
&b_i=\frac{|r_{i+1}|^2d_i}{|r_i|^2}=-\frac{(r_{i+1},Ap_i)}{(p_i,Ap_i)}.
\end{aligned} \tag{9:1}$$

The connections between a_i, b_i, d_i are given by the equation

$$\mu(r_0)-\frac{d_0}{a_0}=0$$

$$\mu(r_i)-\frac{d_i}{a_i}=\frac{b_{i-1}}{a_{i-1}}=\frac{|r_i|^2}{|r_{i-1}|^2}\,\frac{d_{i-1}}{a_{i-1}} \qquad (i>0), \tag{9:2}$$

where $\mu(r)$ is the Rayleigh quotient (4:12). In order to establish these relations we use the fact that r_i and r_{i+1} are orthogonal. This yields

$$\left.\begin{aligned}
|r_i|^2&=a_i(r_i,Ap_i)\\
|r_{i+1}|^2&=-a_i(r_{i+1},Ap_i)
\end{aligned}\right\} \quad (i\geqq 0)$$

by virtue of the formula $r_{i+1}=r_i-a_iAp_i$. From the connection between p_i and r_i, we find that

$$\begin{aligned}
\frac{d_i}{a_i}|r_i|^2&=d_i(r_i,Ap_i)\\
&=(r_i,Ar_i)+b_{i-1}(r_i,Ap_{i-1})\\
&=(r_i,Ar_i)-\frac{b_{i-1}}{a_{i-1}}|r_i|^2.
\end{aligned}$$

This yields (9:2) in case $i>0$. The formula, when $i=0$, follows similarly.

In the formulas (9:1) the scalar factor d_i is an arbitrary positive number determining the length of p_i. The case $d_i=1$ is discussed in sections 3 and 5. The following cases are of interest.

I. *The vector p_i can be chosen to be the residual vector $\bar{r}_i$ described in section 7.*

In this event we select

$$d_0=1, \qquad d_{i+1}=1+b_i. \tag{9:3}$$

The formula (7:2b) for $\bar{x}_{i+1}$ becomes

$$\bar{x}_{i+1}=\frac{x_{i+1}+b_i\bar{x}_i}{1+b_i}. \tag{9:4}$$

II. *The vector p_i can be chosen so that the formula*

$$p_i=\sum_{j=0}^{i}\frac{r_j}{|r_j|^2}$$

holds.

In this event the basic formulas (9:1) take the simple form

$$\begin{aligned}
&r_0=k-Ax_0, \qquad p_0=\frac{r_0}{|r_0|^2}\\
&x_{i+1}=x_i+\frac{p_i}{(p,Ap_i)}\\
&r_{i+1}=r_i-\frac{Ap_i}{(p_i,Ap_i)}\\
&p_{i+1}=p_i+\frac{r_{i+1}}{|r_{i+1}|^2}.
\end{aligned} \tag{9:5}$$

This result is obtained from (9:1) by choosing

$$d_i=|r_i|^2.$$

In this case the formulas (9:5) are very simple and are particularly adaptable to computation. It has the disadvantage that the vectors p_i may grow considerably in length, as can be seen from the relations

$$|p_{i+1}|^2=|p_i|^2+\frac{1}{|r_{i+1}|^2}.$$

However, if "floating" operations are used, this should present no difficulty.

III. *The vector p_i can be chosen to be the correction to be added to x_i in the $(i+1)$st relaxation.*

In this event, $a_i=1$ and the formulas (9:1) take the form

$$\begin{aligned}
&r_0=k-Ax_0, \qquad p_0=\frac{r_0}{d_0}\\
&x_{i+1}=x_i+p_i\\
&r_{i+1}=r_i-Ap_i\\
&p_{i+1}=\frac{r_{i+1}+b_ip_i}{d_{i+1}}\\
&d_0=\mu(r_0), \qquad d_{i+1}=\mu(r_{i+1})-b_i\\
&b_i=\frac{|r_{i+1}|^2}{|r_i|^2}d_i.
\end{aligned} \tag{9:6}$$

These relations are obtained from (9:1) and (9:2) by setting $a_i=1$.

IV. *The vector p_i can be chosen so that a_i is the reciprocal of the Rayleigh quotient of r_i.*

The formulas for a_i, b_i and d_i in (9:1) then become

$$a_i=\frac{|r_i|^2}{(r_i,Ar_i)}$$

$$b_i=\frac{|r_{i+1}|^2}{|r_i|^2}d_i$$

$$d_0=1, \qquad d_{i+1}=1-\frac{b_i a_{i+1}}{a_i}.$$

This is sufficient to indicate the variety of choices that can be made for the scalar factor $d_0, d_1, \ldots$. For purposes of computation the choice $d_i=1$ appears to be the simplest, all things considered.

10. Extensions of the cg-Method

In the preceding pages we have assumed that the matrix A is a positive definite symmetric matrix. The algorithm (3:1) still holds when A is nonnegative and symmetric. The routine will terminate when one of the following situations is met:

(1) The residual r_m is zero. In this event x_m is a solution of $Ax=k$, and the problem is solved.

(2) The residual r_m is different from zero but $(Ap_m,p_m)=0$, and hence $Ap_m=0$. Since $p_i=c_i\overline{r}_i$, it follows that $A\overline{r}_m=0$, where $\overline{r}_m$ is the residual of the vector $\overline{x}_m$ defined in section 7. The point $\overline{x}_m$ is accordingly a point at which $|k-Ax|^2$ attains its minimum. In other words, $\overline{x}_m$ is a least-square solution. One should observe that $p_m\neq 0$ (and hence $\overline{r}_m\neq 0$). Otherwise, we would have $r_m=-b_{m-1}p_{m-1}$, contrary to the fact that r_m is orthogonal to p_{m-1}. The point x_m fails to minimize the function

$$g(x)=(x,Ax)-2(k,x),$$

for in this event

$$g(x_m+tp_m)=g(x_m)-2t|r_m|^2.$$

In fact, $g(x)$ fails to have a minimum value.

It remains to consider the case when A is a general nonsingular matrix. In this event we observe that the matrix A^*A is symmetric and that the system $Ax=k$ is equivalent to the system

$$A^*Ax=A^*k. \tag{10:1}$$

Applying the eq (3:1) to this last system, we obtain the following iteration,

$$\begin{aligned} r_0&=k-Ax_0, \qquad p_0=A^*r_0, \\ a_i&=\frac{|A^*r_i|^2}{|Ap_i|^2} \\ x_{i+1}&=x_i+a_ip_i \\ r_{i+1}&=r_i-a_iAp_i \\ b_i&=\frac{|A^*r_{i+1}|^2}{|A^*r_i|^2} \\ p_{i+1}&=A^*r_{i+1}+b_ip_i. \end{aligned} \tag{10:2}$$

If one does not wish to use any properties of the cg-method in the computation of a_i and b_i besides the defining relations, since they may be disturbed by rounding-off errors, one should use the formulas

$$a_i=\frac{(Ap_i,r_i)}{|Ap_i|^2}$$

$$b_i=-\frac{(Ap_i,AA^*r_{i+1})}{|Ap_i|^2}.$$

In this case the error function $f(x)$ is the function $f(x)=|k-Ax|^2$, and hence is the squared residual. It is a simple matter to interpret the results given above for this new system.

It should be emphasized that, even though the use of the system (10:2) is equivalent from a theoretical point of view to applying the cg-algorithm to the system (10:1), the two methods are not equivalent from a numerical point of view. This follows because rounding-off errors in the two methods are not the same. The system (10:2) is the better of the two, because at all times one uses the original matrix A instead of the computed matrix A^*A, which will contain rounding-off errors.

There is a slight generalization of the system (10:2) that is worthy of note. This generalization consists of selecting a matrix B such that BA is positive definite and symmetric. The matrix B is necessarily of the form A^*H, where H is positive definite and symmetric. We can apply the cg-algorithm to the system

$$BAx=Bk. \tag{10:3}$$

In place of (10:2) one obtains the algorithm

$$\begin{aligned} r_0&=k-Ax_0, \qquad p_0=Br_0, \\ a_i&=\frac{|Br_i|^2}{(p_i,BAp_i)}, \\ x_{i+1}&=x_i+a_ip_i, \\ r_{i+1}&=r_i-a_iAp_i, \\ b_i&=\frac{|Br_{i+1}|^2}{|Br_i|^2}, \\ p_{i+1}&=Br_{i+1}+b_ip_i. \end{aligned} \tag{10:4}$$

Again the formulas for a_i and b_i, which are given directly by the defining relations, are

$$a_i=\frac{(p_i,Br_i)}{(p_i,BAp_i)}$$

$$b_i=-\frac{(Br_{i+1},BAp_i)}{(p_i,BAp_i)}$$

When $B=A^*$, this system reduces to (10:2). If A is symmetric and positive definite, the choice $B=I$ gives the original cg-algorithm.

There is a generalization of the cd-algorithm concerning which a few remarks should be made. In this method we select vectors $p_0, \ldots, p_{n-1}$ and $q_0, \ldots, q_{n-1}$ such that

$$(q_i, Ap_j) = 0 \qquad (i \neq j),$$
$$(q_i, Ap_j) > 0. \tag{10:5}$$

The solution can be obtained by the recursion formulas

$$\begin{aligned} r_0 &= k - Ax_0, \\ a_i &= \frac{(q_i, r_i)}{(q_i, Ap_i)} = \frac{(q_i, r_0)}{(q_i, Ap_i)}, \\ x_{i+1} &= x_i + a_i p_i, \\ r_{i+1} &= r_i - a_i A p_i. \end{aligned} \tag{10:6}$$

The problem is then reduced to finding the vectors p_i, q_i such that (10:5) holds. We shall show in a moment that q_i is of the form

$$q_i = B^* p_i, \tag{10:7}$$

where B has the property that BA is symmetric and positive definite. The algorithm (10:6) is accordingly equivalent to applying the cd-algorithm to (10:3). To see that q_i is of the form (10:7), let P be the matrix whose column vectors are $p_0, \ldots, p_{n-1}$ and Q be the matrix whose column vectors are $q_0, \ldots, q_{n-1}$. The condition (10:5) is equivalent to the statement that the matrix $D = Q^*AP$ is a diagonal matrix whose diagonal terms are positive. Select B so that $Q = B^*P$. Then $D = P^*BAP$ from which we conclude that BA is a positive definite symmetric matrix, as was to be proved.

In view of the results just obtained, we see that the algorithm (10:4) is the *most general cg-algorithm for any linear system*. Similarly, the most general cd-algorithm is obtained: by (i) selecting a matrix B such that BA is symmetric and positive definite, (ii) selecting nonzero vectors $p_0, \ldots, p_{n-1}$ such that

$$(p_i, BAp_j) = 0, \qquad (i \neq j)$$

and (iii), using the recursion formulas

$$\begin{aligned} r_0 &= k - Ax_0 \\ a_i &= \frac{(p_i, Br_i)}{(p_i, BAp_i)} = \frac{(p_i, Br_0)}{(p_i, BAp_i)} \\ x_{i+1} &= x_i + a_i p_i \\ r_{i+1} &= r_i - a_i A p_i. \end{aligned}$$

11. Construction of Mutually Conjugate Systems

As was remarked in section 4 the cd-method is not complete until a method of constructing a set of mutually conjugate vectors $p_0, p_1, \ldots$ has been given. In the cg-method the choice of the vector p_i depended on the result obtained in the previous step. The vectors $p_0, p_1, \ldots$ are accordingly determined by the starting point x_0 and vary with the point x_0.

Assume again that A is a positive definite, symmetric matrix. In a cd-method the vectors $p_0, p_1, \ldots$ can be chosen to be independent of the starting point. This can be done, for example, by starting with a set of n linearly independent vectors $u_0, u_1, \ldots, u_{n-1}$ and constructing conjugate vectors by a successive A-orthogeonalization process. For example, we may use the formulas

$$\begin{aligned} p_0 &= u_0, \\ p_1 &= u_1 - \alpha_{10} p_0, \\ p_2 &= u_2 - \alpha_{20} p_0 - \alpha_{21} p_1, \\ &\cdot \\ &\cdot \\ &\cdot \\ p_i &= u_i - \alpha_{i0} p_0 - \alpha_{i1} p_1 - \ldots - \alpha_{i,i-1} p_{i-1}. \\ &\cdot \\ &\cdot \\ &\cdot \end{aligned} \tag{11:1}$$

The coefficient $\alpha_{ij} (i > j)$ is to be chosen so that p_i is conjugate to p_j. The formula for α_{ij} is evidently

$$\alpha_{ij} = \frac{(u_i, Ap_j)}{(p_j, Ap_j)} \qquad (j < i). \tag{11:2}$$

Observe that

$$(p_i, Au_j) = 0 \qquad (j < i)$$
$$(p_i, Au_i) = (p_i, Ap_i). \tag{11:3}$$

Using (11:3) we see that alternately

$$\alpha_{ij} = \frac{(Au_i, p_j)}{(Au_j, p_j)}. \tag{11:4}$$

As described in section 4, the successive estimates of the solution are given by the recursion formula

$$x_0 = 0, \qquad x_{i+1} = x_i + a_i p_i, \tag{11:5}$$

where

$$a_i = \frac{(p_i, k)}{(p_i, Ap_i)}. \tag{11:6}$$

There is a second method of computing the vectors $p_0, p_1, \ldots, p_{n-1}$, given by the recursion formulas

$$u_i^{(0)} = u_i \tag{11:7a}$$

$$p_j = u_j^{(j)}, \tag{11:7b}$$

$$u_i^{(j+1)} = u_i^{(j)} - \alpha_{ij} p_j, \qquad (i = j+1, \ldots, n) \tag{11:7c}$$

$$\alpha_{ij} = \frac{(u_i^{(j)}, Au_j)}{(p_j, Au_j)} \qquad (i > j). \tag{11:7d}$$

We have the relations (11:3) and,

$$(u_i^{(k)}, Au_j) = 0 \qquad (j<k) \qquad (11{:}8a)$$

$$(u_i^{(k)}, Ap_j) = 0 \qquad (j<k) \qquad (11{:}8b)$$

$$u_i^{(k)} = u_i - \alpha_{i0}p_0 - \ldots - \alpha_{i,j-1}p_{j-1} \qquad (i>j) \qquad (11{:}8c)$$

$$(p_i, Ap_j) = 0 \qquad (i \neq j). \qquad (11{:}8d)$$

The eq (11:8a) hold when $k=j+1$ by virtue of (11:7c) and (11:7d). That they hold for other values of $j<k$ follows by induction. Equation (11:8c) follows from (11:7c).

If one selects successively $u_0=r_0$, $u_1=r_1$, . . ., $u_{n-1}=r_{n-1}$, the procedure just described is equivalent to the cg-method described in section 3, in the sense that the same estimates x_0, x_1, . . . and the same direction vectors p_0, p_1, . . . are obtained. If one selects $u_0=k$, $u_1=Ak$, . . ., $u_{n-1}=A^{n-1}k$, one again obtains the same estimates x_0, x_1, . . . as in the cg-method with $x_0=0$. However in this event the vectors p_0, p_1, . . . are multiplied by nonzero scalar factors. On the other hand if one selects $u_0=(1,0,\ldots,0)$, $u_1=(0,1,\ldots,0)$, . . ., $u_{n-1}=(0,\ldots,0,1)$ the cd-method is equivalent to the Gauss elimination method. This case will be discussed in the next section.

12. Connections With the Gauss Elimination Method [12]

In the present section it will be convenient to use the range 1, . . ., n in place of 0, 1, . . ., $n-1$ used heretofore, except for the notations x_0, x_1, . . ., x_n describing the successive estimates of the solution. Let e_1, . . ., e_n be the unit vectors $(1,0,\ldots,0)$, $(0,1,0,\ldots,0)$, . . ., $(0,\ldots,0,1)$. These vectors will play the role of the vectors u_0, . . ., u_{n-1} of section 11. The eq (11:7), together with (11:4) and (11:5), yield the recursion formulas

$$u_i^{(1)} = e_i \qquad (i=1,\ldots,n) \qquad (12{:}1a)$$

$$p_i = u_i^{(1)} \qquad (12{:}1b)$$

$$u_i^{(j+1)} = u_i^{(j)} - \alpha_{ij}p_j \qquad (i=j+1,\ldots,n) \qquad (12{:}1c)$$

$$\alpha_{ij} = \frac{(Au_i^{(j)}, e_j)}{(Ap_j, e_j)} \qquad (12{:}1d)$$

$$x_0 = 0, \qquad x_i = x_{i-1} + a_i p_i \qquad (12{:}1e)$$

$$a_i = \frac{(p_i, k)}{(Ap_i, e_i)}. \qquad (12{:}1f)$$

These formulas generate mutually conjugate vectors p_1, . . ., p_n and corresponding estimates x_1, . . ., x_n of the solution of $Ax=k$. In particular x_n is the desired solution. The advantage of this method lies in the ease with which the inner products appearing in (12:1d) and (12:1f) can be computed. A systematic scheme for carrying out the computations will now be given. The scheme is that commonly used in elimination. In the presentation that we now give, extraneous entries will be kept so as to give the reader a clear picture of the results obtained.

We begin by writing the matrices A, I and the vector k as a single matrix

$$\begin{array}{ccccccccccccc}
a_{11} & a_{12} & a_{13} & \ldots & a_{1n} & 1 & 0 & 0 & \ldots & 0 & k_1 & \\
a_{21} & a_{22} & a_{23} & \ldots & a_{2n} & 0 & 1 & 0 & \ldots & 0 & k_2 & \\
a_{31} & a_{32} & a_{33} & \ldots & a_{3n} & 0 & 0 & 1 & \ldots & 0 & \cdot & \\
\cdot & \cdot & \cdot & & \cdot & \cdot & & & & \cdot & \cdot & (12{:}2) \\
\cdot & \cdot & \cdot & & \cdot & \cdot & & & & \cdot & \cdot & \\
\cdot & \cdot & \cdot & & \cdot & \cdot & & & & 0 & \cdot & \\
a_{n1} & a_{n2} & a_{n3} & \ldots & a_{nn} & 0 & \ldots & & 0 & 1 & k_n. &
\end{array}$$

The vector p_1 is the vector $(1,0,\ldots,0)$, and $a_1 = k_1/a_{11}$ is defined by (12:1f). Hence,

$$x_1 = a_1p_1 = \left(\frac{k_1}{a_{11}}, 0, \ldots, 0\right)$$

is our first estimate. Observe also that

$$\alpha_{i1} = \frac{a_{i1}}{a_{11}}.$$

Multiplying the first row by α_{i1} and subtracting the result from the ith row $(i=2,\cdots,n)$, we obtain the new matrix

$$\begin{array}{ccccccccc}
a_{11} & a_{12} & a_{13} & \ldots & a_{1n} & p_{11} & \ldots & p_{1n} & k_1 \\
0 & a_{22}^{(2)} & a_{23}^{(2)} & \ldots & a_{2n}^{(2)} & p_{21} & \ldots & p_{2n} & k_2^{(2)} \\
0 & a_{32}^{(2)} & a_{33}^{(2)} & \ldots & a_{3n}^{(2)} & u_{31}^{(2)} & \ldots & u_{3n}^{(2)} & k_3^{(2)} \\
\cdot & & & & \cdot & \cdot & & \cdot & \cdot \\
\cdot & & & & \cdot & \cdot & & \cdot & \cdot \\
\cdot & & & & \cdot & \cdot & & \cdot & \cdot \\
0 & a_{n2}^{(2)} & a_{n3}^{(2)} & \ldots & a_{nn}^{(2)} & u_{n1}^{(2)} & \ldots & u_{nn}^{(2)} & k_n^{(2)}
\end{array} \qquad (12{:}3)$$

One should observe that $(0,k_2^{(2)},\ldots,k_n^{(2)})$ is the residual of x_1. By the procedure just described the ith row $(i>1)$ of the identity matrix has been replaced by $u_i^{(2)}$, the second row yielding the vector $p_2=u_2^{(2)}$. Observe also that

$$a_{i2}^{(2)} = (Au_i^{(2)}, e_i) \qquad (i=2,\ldots,n)$$

$$a_{22}^{(2)} = (Ap_2, e_2), \qquad k_2^{(2)} = (p_2, k).$$

Hence,

$$x_2 = x_1 + \frac{k_2^{(2)}}{a_{22}^{(2)}}p_2$$

is the next estimate of the solution. Moreover,

$$\alpha_{i2} = \frac{a_{i2}^{(2)}}{a_{22}^{(2)}} \qquad (i=3,\ldots,n).$$

[12] cf. Fox, Huskey, and Wilkinson, loc. cit.

Next multiply the 2nd row of (12:3) by α_{i2} and subtract the result from the ith row $(i=3,\cdots,n)$. We obtain

$$\begin{array}{cccccccccc}
a_{11} & a_{12} & a_{13} & \ldots & a_{1n} & p_{11} & \ldots & p_{1n} & k_1 \\
0 & a_{22}^{(2)} & a_{23}^{(2)} & \ldots & a_{2n}^{(2)} & p_{21} & \ldots & p_{2n} & k_2^{(2)} \\
0 & 0 & a_{33}^{(3)} & \ldots & a_{3n}^{(3)} & p_{31} & \ldots & p_{3n} & k_3^{(3)} \\
0 & 0 & a_{43}^{(3)} & \ldots & a_{4n}^{(3)} & u_{41}^{(3)} & \ldots & u_{4n}^{(3)} & k_4^{(3)} \\
\cdot & \cdot & \cdot & & \cdot & \cdot & & \cdot & \cdot \\
\cdot & \cdot & \cdot & & \cdot & \cdot & & \cdot & \cdot \\
\cdot & \cdot & \cdot & & \cdot & \cdot & & \cdot & \cdot \\
0 & 0 & a_{n3}^{(3)} & \ldots & a_{nn}^{(3)} & u_{n1}^{(3)} & \ldots & u_{nn}^{(3)} & k_n^{(3)}.
\end{array}$$

The vector $(0,0,k_3^{(3)},\ldots,k_n^{(3)})$ is the residual of x_2. The elements $u_{i1}^{(3)},\ldots,u_{in}^{(3)}$ form the vector $u_i^{(3)} (i=3)$ and $p_3=u_3^{(3)}$. We have

$$a_{i3}^{(3)}=(Au_i^{(3)}, e_i) \qquad (i=3,\ldots,n)$$

$$a_{33}^{(3)}=(Ap_3, e_3), \qquad k_3^{(3)}=(p_3, k).$$

We have accordingly

$$x_3=x_2+\frac{k_3^{(3)}}{a_{33}^{(3)}}\, p_3,$$

and

$$\alpha_{i3}=\frac{a_{i3}^{(3)}}{a_{33}^{(3)}} \qquad (i=4,\ldots,n)$$

Proceeding in this manner, we finally obtain a matrix of the form

$$\begin{array}{cccccccccc}
a_{11} & a_{12} & a_{13} & \ldots & a_{1n} & p_{11} & \ldots & p_{1n} & k_1 \\
0 & a_{22}^{(2)} & a_{23}^{(2)} & \ldots & a_{3n}^{(2)} & p_{21} & \ldots & p_{2n} & k_1^{(2)} \\
0 & 0 & a_{33}^{(3)} & \ldots & a_{3n}^{(3)} & p_{31} & \ldots & p_{3n} & k_1^{(3)} \\
\cdot & \cdot & & & \cdot & \cdot & & \cdot & \cdot \\
\cdot & \cdot & & & \cdot & \cdot & & \cdot & \cdot \\
\cdot & \cdot & & & \cdot & \cdot & & \cdot & \cdot \\
0 & 0 & \ldots & 0 & a_{nn}^{(n)} & p_{n1} & \ldots & p_{nn} & k_n^{(n)}
\end{array} \tag{12:4}$$

The elements $p_{i1}, \cdots, p_{in}$ define a vector p_i. The vectors $p_1, \cdots, p_n$ are the mutually conjugate vectors defined by the iteration (12:1). At each stage

$$\alpha_{ij}=\frac{a_{ij}^{(j)}}{a_{jj}^{(j)}} \qquad (i=j+1,\ldots,n)$$

$$a_{ii}^{(i)}=(p_i, Ap_i), \qquad k_i^{(i)}=(p_i, k)=(p_i, r_i).$$

Moreover the estimate x_i of the solution h is given by the formula

$$x_i=x_{i-1}+\frac{k_i^{(i)}}{a_{ii}^{(i)}}\, p_i.$$

The vector $0, \cdots, 0, a_{ii}^{(i)}, \cdots, a_{in}^{(i)}$ defined by the first n elements in the ith row of (12:4) is the vector Ap_i. If we denote by P the matrix whose column vectors are $p_1, p_2, \cdots, p_n$, then the matrix (12:4) is the matrix

$$||P^*A \quad P^* \quad P^*k||.$$

The matrices P^*A and P are triangular matrices with zeros below the diagonal. The matrix $D=P^*AP$ is the diagonal matrix whose diagonal elements are $a_{11}, a_{22}^{(2)}, \ldots, a_{nn}^{(n)}$. The determinant of P is unity and the determinant of A is the product

$$a_{11}a_{22}^{(2)} \ldots a_{nn}^{(n)}.$$

As was seen in section 4, if we let

$$f(x)=(h-x, A(h-x)),$$

the sequence

$$f(x_0), f(x_1), \ldots, f(x_{n-1}), f(x_n)=0$$

decreases monotonically. No general statement can be made for the sequence

$$|y_0|, |y_1|, \ldots, |y_{n-1}|, |y_n|=0$$

of lengths of the error vectors $y_i=h-x_i$. In fact, we shall show that this sequence can increase monotonically, except for the last step. A situation of this type cannot arise when the cg-process is used.

If A is nonsymmetric, the interpretation given above must be modified somewhat. An analysis of the method will show that one finds implicitly two triangular matrices P and Q such that Q^*AP is a diagonal matrix. To carry out this process, it may be necessary to interchange rows of A. By virtue of the remarks in section 10, the matrix Q^* is of the form B^*P. The general procedure is therefore equivalent to application of the above process to the system (10:3).

13. An Example

In the cg-method the estimates $x_0, x_1, \ldots$ of the solution h of $Ax=k$ have the property that the error vectors $y_0=h-x_0$, $y_1=h-x_1$, $\ldots$ are decreased in length at each step. This property is not enjoyed by every cd-method. In this section we construct an example such that, for the estimates $x_0=0, x_1, \ldots,$ of the elimination method,

$$|h-x_{i-1}|<|h-x_i| \qquad (i=1,\ldots,n-1).$$

If the order of elimination is changed, this property may not be preserved.

The example we shall give is geometrical instead of numerical. Start with an $(n-1)$-dimensional ellipsoid E_n with center $x_n=h$ and with axes of unequal length. Draw a chord C_n through x_n, which is not orthogonal to an axis of E_n. Select a point $x_{n-1}\neq x_n$ on this chord inside E_n, and pass a hyperplane P_{n-1} through x_{n-1} conjugate to C_n, that is, parallel to the plane determined by the midpoints of the chords of E_n parallel to C_n. Let e_n be a unit vector normal to P_{n-1}. It is clear that e_n is not parallel to C_n. The plane P_{n-1} can be shown to cut E_n in an $(n-2)$-dimensional ellipsoid E_{n-1} with center at x_{n-1} and with axes of unequal length.

Next draw a chord C_{n-1} of E_{n-1} through x_{n-1} which is not orthogonal to an axis of E_{n-1}, and which is not perpendicular to $h-x_{n-1}$. One can then select a point x_{n-2} on C_{n-1} which is nearer to h than x_{n-1}. Let P_{n-2} be the hyperplane through x_{n-2} conjugate to C_{n-1}. It intersects E_{n-1} in an $(n-3)$-dimensional ellipsoid E_{n-2} with center at x_{n-2}. The axes of E_{n-2} can be shown to be of unequal lengths. Let e_{n-1} be a unit vector in P_{n-1} perpendicular to P_{n-2}.

We now repeat the construction made in the last paragraph. Select a chord C_{n-2} of E_{n-2} through x_{n-2} that is not orthogonal to an axis of E_{n-2} and that is not perpendicular to $h-x_{n-2}$. Select x_{n-3} on C_{n-2} nearer to h than x_{n-2}, and let P_{n-3} be a plane through x_{n-3} conjugate to C_{n-2}. It cuts E_{n-2} in an $(n-4)$-dimensional ellipsoid E_{n-3} with center at x_{n-3} with axes of unequal lengths. Let e_{n-2} be a unit vector in P_{n-1} and P_{n-2} perpendicular to P_{n-3}. Clearly, e_n, e_{n-1}, e_{n-2} are mutually perpendicular.

Proceeding in this manner, we can construct

(1) Chords $C_n, C_{n-1}, \ldots, C_1$, which are mutually conjugate.

(2) Planes $P_{n-1}, \ldots, P_1$ such that P_k is conjugate to C_{k+1}. The chords $C_1, \ldots, C_k$ lie in P_k.

(3) The intersection of the planes $P_{n-1}, \ldots, P_k$, which cuts E_n in a $(k-1)$-dimensional ellipsoid E_k with center x_k.

(4) The point x_i, which is closer to h than x_{i+1}, $i<n-1$.

(5) The unit vectors $e_n, \ldots, e_2, e_1$ (with e_1 in the direction of C_1), which are mutually orthogonal.

Let x_0 be an arbitrary point on C_1 that is nearer to h than x_1. Select a coordinate system with x_0 as the origin and with $e_1, \ldots, e_n$ as the axes. In this coordinate system the elimination method described in the last section will yield as successive estimates the points $x_1, \ldots, x_n$ described above. These estimates have the property that x_i is closer to $x_n=h$ than x_{i+1} if $i<n-1$.

As a consequence of the construction just made we see that, given a set of mutually conjugate vectors $p_1, \ldots, p_n$ and a starting point x_0, one can always choose a coordinate system such that the elimination method will generate the vectors $p_1, \ldots, p_n$ (apart from scalar factors) and will generate the same estimates $x_1, \ldots, x_n$ of h as the cd-method determined by these data. One needs only to select the origin at x_0, the vector e_1 parallel to p_1, the vector e_2 in the plane of p_1 and p_2 and perpendicular to e_1, the vector e_3 in the plane of p_1, p_2, p_3 and perpendicular to e_1 and e_2, and so on. This result may have no practicale value, but it does serve to clarify the relationship between the elimination method and the cd-method, and also the relationship between the elimination method and the cg-method.

14. A Duality Between Orthogonal Polynomials and n-Dimensional Geometry

The method of conjugate gradients is related to the theory of orthogonal polynomials and to continued fraction expansions. To develop this, we first study connections between orthogonal polynomials and n-dimensional geometry.

Let $m(\lambda)$ be a nonnegative and nondecreasing function on the interval $0\leq\lambda\leq l$. The (Riemann) Stieltjes integral

$$\int_0^l f(\lambda)dm(\lambda)$$

then exists for any continuous function $f(\lambda)$ on $0\leq\lambda\leq l$. We call $m(\lambda)$ a *mass distribution* on the positive λ-axis. The following two cases must be distinguished.

(a) The function $m(\lambda)$ has infinitely many points of increase on $0<\lambda<l$.

(b) There are only a finite number n of points of increase. In both cases we may construct by orthogonalization of the successive powers $1, \lambda, \lambda^2, \cdots, \lambda^n$ a set of $n+1$ orthogonal polynomials [13]

$$R_0(\lambda), R_1(\lambda), \cdots, R_n(\lambda) \qquad (14:1)$$

with respect to the mass distribution. One has

$$\int_0^l R_i(\lambda)R_k(\lambda)dm(\lambda)=0 \qquad (i\neq k)\cdot \qquad (14:2)$$

The polynomial $R_i(\lambda)$ is of degree i. In case (b), $R_n(\lambda)$ is a polynomial of degree n having its zeros at the n points of increase of $m(\lambda)$. In both cases the zeros of each of the polynomials (14:1) are real and distinct and located inside the interval $(0,l)$. Hence we may normalize the polynomials so that

$$R_i(0)=1 \qquad (i=1, \cdots, n). \qquad (14:3)$$

The polynomials (14:1) are then uniquely determined by the mass distribution.

During the following investigations we use the *Gauss mechanical quadrature as* a basic tool. It can be described as follows: If $\lambda_1, \cdots, \lambda_n$ denote the zeros of $R_n(\lambda)$, there exist *positive* weight coefficients $m_1, m_2, \cdots, m_n$ such that,

$$\int_0^l R(\lambda)dm(\lambda)=m_1R(\lambda_1)+m_2R(\lambda_2)+\ldots+m_nR(\lambda_n) \qquad (14:4)$$

whenever $R(\lambda)$ is a polynomial of degree at most $2n-1$. In the special case b) the λ_k are the abscissas where $m(\lambda)$ jumps and the m_k the corresponding jump.

In order to establish the duality mentioned in the title of this section, we construct a positive definite matrix A having $\lambda_1, \lambda_2, \cdots, \lambda_n$ as eigenvalues (for instance, the diagonal matrix having $\lambda_1, \cdots, \lambda_n$ in the main diagonal and zeros elsewhere). Furthermore, if $e_1, e_2, \cdots, e_n$ are the normalized eigenvectors of A, we introduce the vector

$$r_0=\alpha_1e_1+\alpha_2e_2+\cdots+\alpha_ne_n, \qquad (14:5)$$

[13] The various properties of orthogonal polynomials used in this chapter may be found in G. Szegö, *Orthogonal Polynomials*, American Mathematical Society Colloquium Publications **23** (1939).

where

$$\alpha_i^2 = m_i \qquad (i=1, \cdots, n). \qquad (14:6)$$

We then have

$$A^k r_0 = \alpha_1\lambda_1^k e_1 + \alpha_2\lambda_2^k e_2 + \cdots + \alpha_n\lambda_n^k e_n \qquad (14:7)$$

for $k=0,1, \cdots, n-1$. The vectors $r_0, Ar_0, \cdots, A^{n-1}r_0$ are linearly independent and will be used as a coordinate system. Indeed their determinant is up to the factor $\alpha_1\alpha_2 \cdots \alpha_n$ Van der Monde's determinant of $\lambda_1, \cdots, \lambda_n$. By the correspondence

$$\lambda_k \rightarrow A^k r_0 \qquad (k=0,1, \cdots, n-1) \qquad (14:8)$$

every polynomial of *maximal degree* $n-1$ is mapped onto a vector of the n-dimensional space and a one-one correspondence between these polynomials and vectors is established. The correspondence has the following properties:

Theorem 14:1. *Let the space of polynomials $R(\lambda)$ of degree $\le n-1$ be metrized by the norm*

$$\|R\| = \left[\int_0^l R(\lambda)^2 dm(\lambda)\right]^{\frac{1}{2}}.$$

Then the correspondence described above is isometric, that is,

$$\int_0^l R(\lambda)\, R'(\lambda)\, dm(\lambda) = (r, r'),$$

where $R(\lambda)$, $R'(\lambda)$ are the polynomials corresponding to r and r'.

It is sufficient to prove this for the powers 1, λ, $\lambda^2, \ldots, \lambda^{n-1}$. Let λ^j, λ^k be two of these powers. From Gauss' formula (14:4) follows

$$\int_0^l \lambda^j\lambda^k dm(\lambda) = \int_0^l \lambda^{j+k} dm(\lambda)$$
$$= m_1\lambda_1^{j+k} + m_2\lambda_2^{j+k} + \ldots + m_n\lambda_n^{j+k}.$$

But (14:5), (14:6), and (14:7) show that this is exactly the scalar product $(A^j r_0, A^k r_0)$ of the corresponding vectors.

Theorem 14:2. *Let the space of polynomials $R(\lambda)$ of degree $\le n-1$ be metrized by the norm*

$$\left[\int_0^l R(\lambda)^2 \lambda dm(\lambda)\right]^{\frac{1}{2}}.$$

Then for polynomials $R(\lambda), R'(\lambda)$ corresponding to r, r' one has

$$\int_0^l R(\lambda) R'(\lambda)\, \lambda dm(\lambda) = (Ar, r'), \qquad (14:9)$$

that is, the correspondence is isometric with respect to the weight function $\lambda dm(\lambda)$ and the metric, determined by the matrix A.

Again we may restrict ourselves to the powers $1, \lambda, \ldots, \lambda^{n-1}$. That is, we must show that

$$\int_0^l \lambda^{j+1}\lambda^k dm(\lambda) = (A^{j+1}r_0, A^k r_0) \qquad (j, k \le n-1). \qquad (14:10)$$

If $j < n-1$, this has already been verified. The remaining case

$$\int_0^l \lambda^n\lambda^k dm(\lambda) = (A^n r_0, A^k r_0) \qquad (k \le n-1) \qquad (14:11)$$

follows in the same manner from Gauss' integration formula, since $n+k \le 2n-1$.

Theorem 14:3. *Let A be a positive definite symmetric matrix with distinct eigenvalues and let r_0 be a vector that is not perpendicular to an eigenvector of A. There is a mass distribution $m(\lambda)$ related to A as described above.*

In order to prove this result let $e_1, \ldots, e_n$ be the normalized eigenvectors of A and let $\lambda_1, \ldots, \lambda_n$ be the corresponding (positive) eigenvalues. The vector r_0 is expressible in the form (14:5). According to our assumption no α_k vanishes. The desired mass distribution can be constructed as a step function that is constant on each of the intervals $0 < \lambda_1 < \lambda_2 < \ldots < \lambda_n < l$, and having a jump at λ_k of the amount $m_k = \alpha_k^2 > 0$, the number l being any number greater than λ_n.

We want to emphasize the following property of our correspondence. If A and r_0 are given, we are able to establish the corredence *without computing eigenvalues of A*. This follows immediately from the basic relation (14:8). Moreover, we are able to compute integrals of the type

$$\int_0^l R(\lambda)\, R'(\lambda)\, dm(\lambda), \qquad \int_0^l R(\lambda)\, R'(\lambda)\, \lambda dm(\lambda), \qquad (14:12)$$

where R, R′ are polynomials of maximal degree $n-1$ without constructing the mass distribution. Indeed, the integrals are equal to the corresponding scalar products (r, r'), (Ar, r') of the corresponding vectors, by virtue of theorems 14:1 and 14:2. Finally, the same is true for the construction of the orthogonal polynomials $R_0(\lambda)$, $R_1(\lambda)$, $\ldots$, $R_n(\lambda)$ because the construction only involves the computation of integrals of the type (14:12). The corresponding vectors $r_0, r_1, \ldots, r_{n-1}$ build *an orthogonal basis in the Euclidian n-space.*

15. An Algorithm for Orthogonalization

In order to obtain the orthogonalization of polynomials, the following method can be used. For any three consecutive orthogonal polynomials the recurrence relation holds:

$$R_{i+1}(\lambda) = (d_i - a_i\lambda) R_i(\lambda) - c_i R_{i-1}(\lambda) \quad R_0 = 1,\ c_0 = 0, \qquad (15:1)$$

where a_i, c_i, d_i are real numbers and $a_i\neq 0$. Taking into account the normalization (14:3), we have

$$1=d_i-c_i. \qquad (15:2)$$

Hence

$$R_{i+1}(\lambda)=(1+c_i-a_i\lambda)R_i(\lambda)-c_iR_{i-1}(\lambda).$$

This relation can be written

$$\frac{R_{i+1}-R_i}{a_i\lambda}=-R_i+\frac{c_i}{a_i}\,\frac{R_i-R_{i-1}}{\lambda}.$$

From this equation it is seen by induction that

$$P_i(\lambda)=-\frac{R_{i+1}-R_i}{a_i\lambda} \qquad (15:3)$$

are polynomials of degree i. Introducing the numbers

$$b_{i-1}=\frac{c_ia_{i-1}}{a_i}, \qquad b_{-1}=0 \qquad (15:4)$$

we have

$$P_i(\lambda)=R_i(\lambda)+b_{i-1}P_{i-1}(\lambda) \qquad (15:5a)$$

$$R_{i+1}(\lambda)=R_i(\lambda)-a_i\lambda P_i(\lambda). \qquad (15:5b)$$

Beginning with $R_0=1$, we are able to compute by (15:5) successively the polynomials $P_0=R_0=1$, R_1, P_1, R_2, P_2, . . ., provided that we know the numbers a_i, b_i. In order to compute them, observe first the relation

$$\int_0^l P_i(\lambda)P_k(\lambda)\lambda dm(\lambda)=0 \qquad (i\neq k). \qquad (15:6)$$

Indeed this integral is up to a constant factor

$$\int_0^l (R_{i+1}-R_i)P_kdm(\lambda).$$

For $k<i$ this is zero, because the second factor is of degree $k<i$.

Using (15:5a) and (15:6), we obtain

$$\int_0^l R_i(\lambda)P_i(\lambda)\lambda dm(\lambda)=\int_0^l P_i(\lambda)^2\lambda dm(\lambda).$$

Combining this result with the orthogonality of R_{i+1} and R_i, we see, by (15:5b), that

$$a_i=\frac{\int_0^l R_i(\lambda)^2dm(\lambda)}{\int_0^l P_i(\lambda)^2\lambda dm(\lambda)}. \qquad (15:7)$$

Using (15:6) and (15:5a),

$$0=\int_0^l R_i(\lambda)P_{i-1}(\lambda)\lambda dm(\lambda)+b_{i-1}\int_0^l P_{i-1}(\lambda)^2\lambda dm(\lambda).$$

Applying (15:3) to the first term yields

$$\frac{1}{a_{i-1}}\int_0^l R_i(\lambda)^2dm(\lambda)=b_{i-1}\int_0^l P_{i-1}(\lambda)^2\lambda dm(\lambda).$$

Combining this result with (15:7), we obtain

$$b_{i-1}=\frac{\int_0^l R_i(\lambda)^2dm(\lambda)}{\int_0^l R_{i-1}(\lambda)^2dm(\lambda)}. \qquad (15:8)$$

The formulas (15:5), (15:7), (15:8), together with $R_0=1$, $b_{-1}=0$, completely determine the polynomials R_0, R_1, . . ., R_{n-1}.

16. A New Approach to the cg-Method, Eigenvalues

In order to solve the system $Ax=k$, we compute the residual vector $r_0=k-Ax_0$ of an initial estimate x_0 of the solution h and establish the correspondence based on A, r_0 described in Theorem 14:3. Without computing the mass distribution, the orthogonalization process of the last section may be carried out by (15:5), (15:7) and (15:8) with $R_0=1$, $b_{-1}=0$. The vectors r_i, p_i corresponding to the polynominals R_i, P_i are therefore determined by the recurrence relations

$$p_i=r_i+b_{i-1}p_{i-1}, \qquad r_{i+1}=r_i-a_iAp_i. \quad (16:1)$$

Multiplication by λ in the domain of polynominals is mapped by our correspondence into applying A in the vector space according to (14:11). In fact,

$$p_i=P_i(A)r_0, \quad r_i=R_i(A)r_0 \qquad (i=0,\ 1,\ .\ .\ .\ ,n-1).$$

The numbers a_i, b_i are computed by (15:7) and (15:8). Using the isometric properties described in theorems 14:1 and 14:2, we find that

$$a_i=\frac{|r_i|^2}{(Ap_i,p_i)}, \qquad b_{i-1}=\frac{|r_i|^2}{|r_{i-1}|^2}.$$

The vectors r_i are orthogonal, and the p_i are conjugate; the latter result follows from (15:6). Hence the basic formulas and properties of the cg-method listed in sections 3 and 5 are established. It remains to prove that the method gives the exact solution after n steps. If we set $x_{i+1}=x_i+a_ip_i$, the corresponding residual is r_{i+1} as follows by induction:

$$k-Ax_{i+1}=(k-Ax_i)-a_iAp_i=r_i-a_iAp_i=r_{i+1}.$$

For the last residual r_n we have ($i=0,\ 1,\ .\ .\ .\ .,n-1$)

$$\begin{aligned}(r_n,r_i)&=(r_{n-1},r_i)-a_{n-1}(Ap_{n-1},r_i)\\&=\int R_{n-1}R_idm-a_{n-1}\int P_{n-1}R_i\lambda dm\\&=\int R_nR_idm=0.\end{aligned}$$

Our basic method reestablishes also the methods of C. Lanczos for computing the *characteristic polynomial* of a given matrix A. Indeed the polynomials R_i, computed by the recurrence relation (15:5), lead finally to the polynomial $R_n(\lambda)$, which, by the basic definition of the correspondence in section 14, is the characteristic polynomial of A, provided that r_0 satisfies the conditions given in theorem 14:3. It may be remembered that orthogonal polynomials build a Sturmian sequence. Therefore, the polynomials $R_0, R_1, \ldots, R_n$ build a *Sturmian sequence for the eigenvalues of the given matrix* A.

Our correspondence allows us to translate every method or result in the vector-space into an analogous method or result for polynomials and vice versa. Let us take as an example the smoothing process in section 7. It is easy to show that the vector $\overline{r}_i$ introduced in that section corresponds to a polynomial $\overline{R}_i(\lambda)$ characterized by the following property: $\overline{R}_i(\lambda)$ is the polynomial of degree i with $\overline{R}_i(0)=1$ having the least-square integral on $(0,l)$. In other words, if r_0 is given by (14:5), then

$$\alpha_1^2\overline{R}_i(\lambda_1)^2+\alpha_2^2\overline{R}_2(\lambda_2)^2+\ .\ .\ .+\alpha_n^2\overline{R}_i(\lambda_n)^2=\text{minimum}.$$

This result may be used to estimate a single eigenvalue of A. In order to compute, for instance, the lowest eigenvalue λ_1, we select r_0 near to the corresponding eigenvector. The first term in the expansion being dominant, the smallest root of $\overline{R}_i(\lambda)$ will be a good approximation of λ_1, provided that i is not too small. Hence the last residual vanishes, being orthogonal to $r_0, r_1, \ldots, r_{n-1}$. It follows that x_n is the desired solution.

17. Example, Legendre Polynomials

Any known set of orthogonal polynomials yields an example of a cg-algorithm. Take, for instance, the Legendre polynomials. Adapted to the interval (0,1), they satisfy the recurrence relation

$$R_{i+1}(\lambda)=\frac{2i+1}{i+1}(1-2\lambda)R_i(\lambda)-\frac{i}{i+1}R_{i-1}(\lambda),\quad R_i(0)=1.$$

From (15:1) and (15:4)

$$a_i=\frac{4i+2,}{i+1}\qquad c_i=\frac{i}{i+1},\qquad b_{i-1}=\frac{2i-1}{2i+1}.\qquad (17:1)$$

This gives the following result, let A be a symmetric matrix having the roots of the Legendre polynomial $R_n(\lambda)$ as eigenvalues, and let

$$r_0=\alpha_1e_1+\alpha_2e_2+\ .\ .\ .+\alpha_ne_n,$$

where $e_1, \ldots, e_n$ are the normalized eigenvectors of A, and $m_1=\alpha_1^2$, $m_2=\alpha_2^2, \ldots, m_n=\alpha_n^2$ are the weight-coefficients for the Gauss' mechanical quadrature with respect to R_n. The cg-algorithm applied to A,r_0 yields the numbers a_i, b_i given by (17:1). Moreover,

$$(r_i,r_i)=b_{i-1}b_{i-2}\ .\ .\ .\ b_0(r_0,r_0)=\frac{1}{2i+1}(r_0,r_0)\qquad (i<n).$$

Hence the residuals decrease during the alogrithm. It may be worth noting that the Rayleigh quotient of r_i is

$$\frac{(r_i,Ar_i)}{|r_i|^2}=\frac{1}{a_i}+\frac{b_{i-1}}{a_{i-1}}=\frac{1}{2}.$$

All residual vectors have the same Rayleigh quotient. This shows that, unlike many other relaxation methods, the cg-process does not necessarily have the tendency of smoothing residuals.

The Chebyshev polynomials yield an example where a_i, b_i are constant for $i>0$.

18. Continued Fractions

Suppose that we have given a mass distribution of type (b) as described in section 14. The function $m(\lambda)$ is a step function with jumps at $0<\lambda_1<\lambda_2<\ldots<\lambda_n<l$, the values of the jumps being $m_1, m_2, \ldots, m_n$, respectively. It is well known[14] that the orthogonal polynominals $R_0(\lambda), R_1(\lambda), \ldots, R_n(\lambda)$, corresponding to this mass distribution, can be constructed by expanding the rational function

$$F(\lambda)=\frac{m_1}{\lambda-\lambda_1}+\cdots+\frac{m_n}{\lambda-\lambda_n}\qquad (18:1)$$

in a continued fraction. The polynominal $R_i(\lambda)$ is the denominator of the ith convergent. For our purposes it is convenient to write the continued fraction in the form

$$F(\lambda)=\cfrac{-1}{d_0-a_0\lambda-\cfrac{c_1}{d_1-a_1\lambda-\cfrac{c_2}{d_2-a_2\lambda-\cfrac{\ddots}{\ \ -\cfrac{c_{n-1}}{d_{n-1}-a_{n-1}\lambda}}}}}\qquad (18:2)$$

[14] H. S. Wall, Analytic Theory of Continued fractions, Van Norstrand (1948).

The denominators of the convergents are given by the recursion formulas

$$R_{i+1}=(d_i-a_i\lambda)R_i-c_iR_{i-1}, \quad R_0=1, \quad c_0=0. \qquad (18:3)$$

This coincides with (15:1). However, in order to satisfy (14:3), the expansion must be carried out so that $d_i=c_i+1$, by virtue of (15:2). The numbers b_i are then given by (15:4). It is clear that

$$F(\lambda)=\frac{Q_{n-1}(\lambda)}{R_n(\lambda)}$$

where

$$Q_{n-1}(\lambda)=\sum_{i=1}^{n} m_i \prod_{j\neq i}(\lambda-\lambda_j).$$

Let us translate these results into the n-dimensional space given by our correspondence. As before, we construct a positive definite symmetric matrix A with eigenvalues $\lambda_1, \ldots, \lambda_n$. Let $e_1, \ldots, e_n$ be corresponding eigenvectors of unit length and choose, as before,

$$r_0=\alpha_1e_1+\ldots+\alpha_ne_n, \qquad \alpha_i^2=m_i.$$

The eigenvalues are the reciprocals of the squares of the semiaxis of the $(n-1)$-dimensional ellipsoid $(x,Ax)=1$. The hyperplane, $(r_0,x)=0$, cuts this ellipsoid in an $(n-2)$-dimensional ellipsoid, E_{n-2}, the squares of whose semiaxis are given by the reciprocals of the zeros of the numerator $Q_{n-1}(\lambda)$ of $F(\lambda)$.

This follows from the fact that if λ_0 is a number such that there is a vector $x_0\neq 0$ orthogonal to r_0 having the property that $(Ax_0,x)=\lambda_0(x_0,x)$ whenever $(r_0,x)=0$, then λ_0 is the square of the reciprocal of the semiaxis of E_{n-2} whose direction is given by x_0. If the coordinate system is chosen so that the axes are given by $e_1, \ldots, e_n$, respectively, then $\lambda=\lambda_0$ satisfies the equation

$$Q_{n-1}(\lambda)=\begin{vmatrix} \lambda_1-\lambda & 0 & 0 & \ldots & 0 & \alpha_1 \\ 0 & \lambda_2-\lambda & 0 & \ldots & 0 & \alpha_2 \\ 0 & 0 & & \ldots & & \\ \cdot & \cdot & & & \cdot & \cdot \\ \cdot & \cdot & & & \cdot & \cdot \\ \cdot & \cdot & & & \cdot & \cdot \\ 0 & 0 & & \ldots & \lambda_n-\lambda & \alpha_n \\ \alpha_1 & \alpha_2 & & \ldots & \alpha_n & 0 \end{vmatrix}=0$$

as was to be proved.

Let us call the zeros of $Q_{n-1}(\lambda)$ the *eigenvalues of A* with respect to r_0 and the polynomial $Q_{n-1}(\lambda)$ the *characteristic polynomial of A with respect to r_0*. The rational function $F(\lambda)$ is accordingly the quotient of this polynomial and the characteristic polynomial of A. Hence we have,

Theorem 18:1. *The numbers a_i, b_i connected with the cg-process of a matrix A and a vector x_0 can be computed by expanding into a continued fraction the quotient built by the characteristic polynomial of A with respect to r_0 and the ordinary characteristic polynomial of A.*

This is the simplest form of the relation between a matrix A, a vector r_0 and the numbers a_i, b_i of the corresponding cg-process. The theorem may be used to investigate the behavior of the a_i, b_i if the eigenvalues of A and those with respect to r_0 are given. The following special case is worth recording. If $m_1=m_2=\ldots=m_n=1$, the rational function is the logarithmic derivative of the characteristic polynomial. From theorem (18:1) follows

Theorem 18:2. *If the vector r_0 of a cg-process is the sum of the normalized eigenvectors of A, the numbers a_i, b_i may be computed by expanding the logarithmic derivative of the characteristic polynomial of A into a continued fraction.*

Finally, we are able to prove

Theorem 18:3. *There is no restriction whatever on the positive constants a_i, b_i in the cg-process, that is, given two sequences of positive numbers $a_0, a_1, \ldots, a_{n-1}$ and $b_0, b_1, \ldots, b_{n-1}$, there is a symmetric positive definite matrix A and a vector r_0 such that the cg-algorithm applied to A, r_0 yield the given numbers.*

The demonstration goes along the following lines: From (15:2) and (15:4), we compute the numbers c_i, d_i, the c_i being again positive. Then we use the continued fraction (18:2) to compute $F(\lambda)$ which we decompose into partial fractions to obtain (18:1). We show next that the numbers λ_i, m_i appearing in (18:1) are positive. After this has been established, our correspondence finishes the proof.

In order to prove that $\lambda_i>0$, $m_i>0$ we observe that the ratio R_{i+1}/R_i is a decreasing function of λ, as can be seen from (18:3) by induction. Using this result, it is not too difficult to show that the polynomials $R_0(\lambda)$, $R_1(\lambda)$, $\ldots$, $R_n(\lambda)$ build a Sturmian sequence in the following sense. The number of zeros of $R_n(\lambda)$ in any interval $a\leq\lambda\leq b$ is equal to the increase of the number of variations in sign in going from a to b. At the point λ_0 there are no variations in sign since $R_i(0)=1$ for every i. At $\lambda=+\infty$, there are exactly n variations because the coefficient of the highest power of λ in $R_i(\lambda)$ is $(-1)^ia_0a_1\ldots a_{i-1}$. Therefore, the roots $\lambda_1, \lambda_2, \ldots, \lambda_n$ of $R_n(\lambda)$ are real and positive.

That the function $F(\lambda)$ is itself a decreasing function of λ follows directly from (18:2). Therefore, its residues $m_1, m_2, \ldots, m_n$ are positive.

In view of theorem 18:3 the numbers a_i in a cg-process can increase as fast as desired. This result was used in section 8.2. Furthermore, the formula

$$b_i=\frac{|r_{i+1}|^2}{|r_i|^2}$$

shows that there is no restriction at all on the behavior of the length of the residual vector during the cg-process. Hence, there are certainly examples where the residual vectors increase in length during the computation, as was stated earlier. This holds in spite of the fact that the error vector $h-x$ decreases in length at each step.

19. Numerical Illustrations

A number of numerical experiments have been made with the processes described in the preceding sections. A preliminary report on these experiments will be given in this section. In carrying out these experiments, no attempt was made to select those which favored the method. Normally, we selected those which might lead to difficulties.

In carrying out these experiments three sets of formulas for a_i, b_i were used in the symmetric case, namely,

$$a_i=\frac{(p_i,r_i)}{(p_i,Ap_i)}, \qquad b_i=-\frac{(r_{i+1},Ap_i)}{(p_i,Ap_i)}, \qquad (19:1)$$

$$a_i=\frac{|r_i|^2}{(p_i,Ap_i)}, \qquad \frac{|r_{i+1}|^2}{|r_i|^2}, \qquad (19:2)$$

$$a_i=\frac{|r_i|^2}{(p_i,Ap_i)d_i}, \quad b_i=\frac{|r_{i+1}|^2}{|r_i|^2}d_i, \quad d_i=1-b_{i-1}\frac{(p_{i-1},Ap_i)}{(p_i,Ap_i)}. \qquad (19:3)$$

In the nonsymmetric case, we have used only the formulas

$$a_i=\frac{|A^*r_i|^2}{|Ap_i|^2}, \qquad b_i=\frac{|A^*r_{i+1}|^2}{|A^*r_i|^2}. \qquad (19:4)$$

Our experience thus far indicates that the best results are obtained by the use of (19:1). Formulas (19:2) were about as good as (19:1) except for very ill conditioned matrices. Most of our experiments were carried out with the use of (19:2) because they are somewhat simpler than (19:1). Formulas (19:3) were designed to improve the relations

$$(r_i,r_{i+1})=0, \qquad (p_i,Ap_{i+1})=0, \qquad (19:5)$$

which they did. Unfortunately, they disturbed the first of the relations

$$(p_i,r_{i+1})=0, \qquad (p_i,Ap_{i+1})=0. \qquad (19:6)$$

A reflection of the geometrical interpretation of the method will convince one that one should strive to satisfy the relations (19:6) rather than (19:5). It is for this reason that (19:1) appears to be considerably superior to (19:3). In place of (19:2), one can use the formulas

$$a_i=\frac{|r_i|^2}{(p_i,Ap_i)}, \qquad b_i=\frac{|r_{i+1}|^2-(r_{i+1},r_i)}{|r_i|^2} \qquad (19:2')$$

to correct rounding off errors. A preliminary experiment indicates that this choice is better than (19:2) and is perhaps as good as (19:1).

A sufficient number of experiments have not been carried out as yet so as to determine the "best" formulas to be used. Our experiments do indicate that floating operations should be used whenever possible. We have also observed that the results in the $(n+1)$st and $(n+2)$nd iterations are normally far superior to those obtained in the nth iteration.

Example 1. This example was selected to illustrate the method of conjugate gradients in case there are no rounding off errors. The matrix A was chosen to be the matrix

$$A=\begin{Vmatrix} 1 & 2 & -1 & 1 \\ 2 & 5 & 0 & 2 \\ -1 & 0 & 6 & 0 \\ 1 & 2 & 0 & 3 \end{Vmatrix}.$$

If we select k to be the vector $(0,2,-1,1)$, the computation is simple. The results at each step are given in table 1.

Normally, the computation is not as simple as indicated in the preceding case. For example, if one selects the solution h to be the vector $(1,1,1,1)$, then k is the vector $(3,9,5,6)$. The results with $(0,0,0,0)$ as the initial estimate is given by table 2.

Table 1.

Step	Vector	Components of the vector				a_i	b_{i-1}
		1	2	3	4		
0	x_0	1	0	0	0		
	r_0	−1	0	0	0		
	p_0	−1	0	0	0		
	Ap_0	−1	−2	1	−1	1	
1	x_1	0	0	0	0		
	r_1	0	2	−1	1		6
	p_1	−6	2	−1	1		
	Ap_1	0	0	0	1	6	
2	x_2	−36	+12	−6	6		
	r_2	0	2	−1	−5		5
	p_2	−30	12	−6	0		
	Ap_2	0	0	−6	−6	5/6	
3	x_3	−61	22	−11	6		
	r_3	0	2	4	0		2/3
	p_3	−20	10	0	0		
	Ap_3	0	10	20	0	1/5	
4	x_4	−65	24	−11	6		

Table 2.

Step	Vector	α times components of vector				α
		1	2	3	4	
0	x_0	0	0	0	0	1
	r_0	3	9	5	6	1
	p_0	3	9	5	6	1
	Ap_0	22	63	27	39	1
1	x_1	453	1359	755	906	β_1
	r_1	−316	−495	933	123	β_1
	p_1	−1935	−2799	6461	1140	$\beta_1\gamma_1$
	Ap_1	−12854	−15585	40701	−4113	$\beta_1\gamma_1$
2	x_2	131702	419553	298277	304149	β_2
	r_2	1689	−34360	−27345	73483	β_2
	p_2	−116022	−1684085	−381080	3066641	$\beta_2\gamma_2$
	Ap_2	−66471	−2579187	−2140458	5685731	$\beta_2\gamma_2$
3	x_3	27589274	84526651	62344884	73103513	β_3
	r_3	542343	−188185	92550	−66019	β_3
	p_3	41725242	−15212135	6969632	−3788997	$\beta_3\gamma_3$
	Ap_3	542343	−188185	92550	−66019	$\beta_3\gamma_3$
4	x_4	1	1	1	1	1
	r_4	0	0	0	0	1

$$\beta_1=1002,\quad \beta_2=326123,\quad \beta_3=69314516,$$

$$\gamma_1=\beta_1/151,\quad \gamma_2=\beta_2/8149,\quad \gamma_3=\beta_3/899615$$

$$a_0=1/\gamma_1,\quad a_1=\gamma_1/\gamma_2,\quad a_2=\gamma_2/\gamma_3,\quad a_3=\gamma_3$$

$$b_0=8149/\beta_1^2,\quad b_1=899615\beta_1\gamma_1/\beta_2^2,\quad b_2=380689\beta_2\gamma_2/\beta_3^2.$$

Table 3.

		k	x_i	r_i
1 2 −1 1	1 0 0 0	3	3	0
2 5 0 2	0 1 0 0	9	0	3
−1 0 6 0	0 0 1 0	5	0	8
1 2 0 3	0 0 0 1	6	0	3
1 2 −1 1	1 0 0 0	3	−3	0
0 1 2 0	−2 1 0 0	3	3	0
0 2 5 1	1 0 1 0	8	0	2
0 0 1 2	−1 0 0 1	3	0	3
1 2 −1 1	1 0 0 0	3	7	0
0 1 2 0	−2 1 0 0	3	−1	0
0 0 1 1	5 −2 1 0	2	2	0
0 0 1 2	−1 0 0 1	3	0	1
1 2 −1 1	1 0 0 0	3	1	0
0 1 2 0	−2 1 0 0	3	1	0
0 0 1 1	5 −2 1 0	2	1	0
0 0 0 1	−6 2 −1 1	1	1	0

The system just described is particularly well suited for elimination. In case k is the vector (3, 9, 5, 6) the procedure described in section 12 yields the results given in table 3. In this table, we start with the matrices A and I. These matrices are transformed into the matrices P^*A and P^* given at the bottom of the table.

It is of interest to compare the error vectors $y_i=h-x_i$ obtained by the two methods just described with $k=(3, 9, 5, 6)$. The error $|y_i|$ is given in the following table.

$\lvert y_i\rvert$	cg-method	Elimination method
$\lvert y_0\rvert$	2. 0	2. 00
$\lvert y_1\rvert$	0. 7	2. 65
$\lvert y_2\rvert$	. 67	4. 69
$\lvert y_3\rvert$	. 65	6. 48
$\lvert y_4\rvert$	. 0	0. 00

In the cg-method $|y_i|$ decreases monotonically, while in the elimination method $|y_i|$ increases except for the last step.

Example 2. In this case the matrix A was chosen to be the matrix

$$\begin{array}{cccccc}
.263879 & .014799 & .016836 & .079773 & -.020052 & .011463\\
-.014799 & .249379 & .028764 & .057757 & -.056648 & -.134493\\
.016836 & .028764 & .263734 & -.033628 & -.012128 & .084932\\
.079773 & .057757 & -.033628 & .215331 & .090696 & -.037489\\
-.020052 & -.056648 & -.012128 & .090696 & .324486 & -.022484\\
.011463 & -.134493 & .084932 & -.037489 & -.022484 & .339271
\end{array}$$

This matrix is a well-conditioned matrix, its eigenvalues lying on the range $\lambda_1=.6035\leqq\lambda\leqq\lambda_6=4.7357$. The computations were carried out on an IBM card programmed calculator with about seven significant figures. The results for the case in which x_0 is the origin and h the vector (1,1,1,1,1,1) are given in table 4.

Example 3. A good illustration of the effects of rounding can be obtained by study of an ill-conditioned system of three equations with three unknowns, namely, the system

$$6x+13y-17z=1$$

$$13x+29y-38z=2$$

$$-17x-38y+50z=-3,$$

whose solution is $x=1$, $y=-3$, $z=-2$. The system was constructed by E. Stiefel. The eigenvalues of A are given by the set

$$\lambda_1=.0588,\qquad \lambda_2=.2007,\qquad \lambda_3=84.7405.$$

The ratio of the largest to the smallest eigenvalue is very large: $\lambda_3/\lambda_1=1441$. The formulas (19:1), (19:2), and (19:3) were used to compute the solution,

TABLE 4.

Starting vector $k=(3.371, 1.2996, 3.4851, 3.7244, 3.0387, 2.412)$

Step	x_i	r_i	p_i	a_i, b_i
0	0 0 0 0 0 0	3.37100 1.29960 3.48510 3.72440 3.03870 2.41200	3.37100 1.29960 3.48510 3.72440 3.03870 2.41200	$a_0=3.092387$ $b_0=0.02360156$
1	1.042444 .4018866 1.077728 1.151729 .9396836 .7458837	−0.3176047 1.011922 .2194351 −.02954774 −.3199108 .03016107	−0.02380454 .1042594 .03016873 .005835219 −.02481941 .008708692	$a_1=3.487517$ $b_1=0.1411714$
2	.9594250 .7654931 1.1829418 1.1720790 .8531255 .7762554	−0.009951160 .004267497 −.01781102 −.009187803 .01514192 .03244676	−0.1331168 .1898594 −.1355206 −.08364038 .1163813 .3367617	$a_2=5.448597$ $b_2=0.3997728$
3	.8868953 .8689395 1.1091023 1.1265069 .9165367 .9597427	.1476560 .1042268 −.1643885 −.0091902 .0633472 −.0907231	.009443967 .01801273 −.02185659 −.004262730 .01098733 .004390484	$a_3=4.580482$ $b_3=0.3769145$
4	.930153 .951447 −1.008989 −1.106982 .966864 .979853	.08593514 .00050406 .05108757 −.12107954 −.03445640 .03607941	.1215308 .06839666 −.03129307 −.1371464 .006956427 .05262778	$a_4=5.464933$ $b_4=0.2541540$
5	.996569 .988825 .991887 1.032032 .970666 1.008614	−.002365634 .000616167 .002508661 .003267702 .006006834 .003155791	.007231114 .02354492 .01713337 −.06753326 .06183634 −.01818237	$a_5=4.742589$ $b_5=0$
6	.999998 .999991 1.000013 1.000004 .999992 .999991	−.00000252 −.00000084 −.00002271 .00000645 .00001636 .00000825		

keeping five significant figures at all times. For comparison, the computation was carried out also with 10 digits, using (19:2). The results are given in table 5. In the third iteration, formula (19:1) gave the better result. In the fourth iteration, formulas (19:1) and (19:2) were equally good, and superior to (19:3). The solution was also carried out by the elimination method using only five significant figures. The results are

	cg-method (19:1)	Elimination
x	.99424	1.00603
y	−2.99518	−3.00506
z	−1.99328	−2.00180

TABLE 5

$x_0=(1, 0, 0)$

Case.	1 Formula (19:2)	2 Formula (19:3)	3 Formula (19:1)	1 with 10 digits
p_0 r_0	5 11 −14	5 11 −14	5 11 −14	5 11 −14
c_0	.011804	.011804	.011804	.01180409347
x_1	.94098 −.12984 .16526	.94098 −.12984 .16526	.94098 −.12984 .16526	.9409795326 −.1298450282 .1652573086
r_1	.14856 .18754 .20021	.14856 .18754 .20021	.14856 .18754 .20021	.1485175838 .1874503815 .2003244444
r_1^2	.097325	.097325	.097325	.09732500125
b_0	.00028458	.00028458	.00027639	.0002845760270
p_1	.14998 .19067 .19623	.14998 .19067 .19623	.14994 .19058 .19634	.1499404639 .1905807178 .1963403800
a_1	7.0058	7.0393	7.0059	7.006740263
x_2	−.10975 −1.46564 −1.20949	−.11477 −1.47202 −1.21606	−.10948 −1.46502 −1.21028	−.1096143529 −1.4651946170 −1.2104487372
r_2	−.15045 .030400 .085455	−.15188 .029648 .084906	−.12747 .081611 .018197	−.1275876043 .0814215368 .0184025802
r_2^2	.030682	.031156	.023240	.02324671838
b_1	.31710	.31860	.23870	.2388565947
p_2	−.10289 .090861 .14768	−.10387 .090685 .14772	−.091679 .12710 .065039	−.0917733357 .1269429981 .0652997748
a_2	.047688	.047713	12.039	12.09069098
x_3	−.10484 −1.46997 −1.21653	−.05079 −1.34651 −1.38837	.99424 −2.99518 −1.99328	.9999886893 −3.000023179 −1.999968135
r_3	−.057616 .23615 −.18543	−.058572 .23643 −.18733	−.086092 −.19036 .25063	−.0009108898 −.0020300857 .0026663150
r_3^2	.093471	.094422	.10646	.000012060204
b_2	3.0287	3.0306	4.5804	.000518791676
p_3	−.36924 .51134 .26185	−.37336 .51126 .26035	−.50602 .39181 .54853	−.0009585010 −.0019642287 .0027001920
a_3	2.9923	2.9762	.011854	.0118007358
x_4	1.00004 −3.00005 −2.00006	1.06040 −2.86812 −2.16322	1.00024 −2.99982 −1.99978	1.0000000003 −2.9999999997 −1.9999999993
r_4	.00064408 .0014340 −.0018823	.00014843 −.00035647 .00094441	.00005181 .0000152 .0000364	0 .0000000008 .0000000002

In this case the results by the cg-method and elimination method appear to be equally effective. The cg-method has the advantage that an improvement can be made by taking one additional step.

This example is also a good illustration for the fact that the size of the residuals is not a reliable criterion for how close one is to the solution. In step 3 the residuals in case 1 are smaller than those of case 3, although the estimate in case 1 is very far from the right solution, whereas in case 3 we are close to it.

Further examples. The largest system that has been solved by the cg-method is a linear, symmetric system of 106 difference equations. The computation was done on the Zuse relay-computer at the Institute for Applied Mathematics in Zurich. The estimate obtained in the 90th step was of sufficient accuracy to be acceptable. [15]

Several symmetric systems, some involving as many as twelve unknowns, have been solved on the IBM card programed calculator. In one case, where the ratio of the largest to the smallest eigenvalue was 4.9, a satisfactory solution has been obtained already in the third step; in another case, where this ratio was 100, one had to carry out fifteen steps in order to get an estimate with six correct digits. In these computations floating operations were not used. At all times an attempt was made to keep six or seven significant figures.

The cg-method has also been applied to the solution of small nonsymmetric systems on the SWAC. The results indicate that the method is very suitable for high speed machines.

A report on these experiments is being prepared at the National Bureau of Standards, Los Angeles.

Los Angeles, May 8, 1952.

[15] See U. Hochstrasser, "Die Anwendung der Methode der konjugierten Gradienten und ihrer Modifikationen auf die Lösung linearer Randwertprobleme," Thesis E. T. H., Zurich, Switzerland, in manuscript.