

A METHOD OF ACCELERATING STATIONARY ITERATIVE METHODS FOR SOLVING LINEAR SYSTEMS

G. K. ROBINSON¹

(Received 5 May 1987; revised 3 August 1987)

Abstract

The speed of convergence of stationary iterative techniques for solving simultaneous linear equations may be increased by using a method similar to conjugate gradients but which does not require the stationary iterative technique to be symmetrisable. The method of refinement is to find linear combinations of iterates from a stationary technique which minimise a quadratic form. This basic method may be used in several ways to construct refined versions of the simple technique. In particular, quadratic forms of much less than full rank may be used. It is suggested that the method is likely to be competitive with other techniques when the number of linear equations is very large and little is known about the properties of the system of equations. A refined version of the Gauss-Seidel technique was found to converge satisfactorily for two large systems of equations arising in the estimation of genetic merit of dairy cattle.

1. Introduction

Consider solving the linear system

$$Au = b \tag{1}$$

where A is a given real $N \times N$ matrix. It is intended to accelerate a linear stationary method of first degree

$$w^{(n+1)} = Gw^{(n)} + k \quad n = 0, 1, 2, \dots \tag{2}$$

where G is a real $N \times N$ matrix and $w^{(n)}$ and k are real $N \times 1$ vectors. Assume that the stationary method is completely consistent in the sense that

$$u = Gu + k$$

¹CSIRO Division of Mathematics and Statistics, Private Bag 10, Clayton, Vic. 3168, Australia.
© Copyright Australian Mathematical Society 1988, Serial-fee code 0334-2700/88

has a unique solution which is the same as the solution of (1). It follows that $I - G$ is non-singular.

The stationary method is said to be symmetrisable if there is a symmetric positive definite matrix H such that $H(I - G)$ is symmetric positive definite. Note that the Gauss-Seidel method is not generally symmetrisable even when A is symmetric positive definite.

Define the error in equation (1) at x to be

$$\varepsilon(x) = x - \bar{u},$$

where \bar{u} is the solution of (1), and define the pseudoresidual associated with (2) at x to be

$$\delta(x) = Gx + k - x.$$

Note that $\delta(x)$ is the step taken by the basic stationary method when started at x . Also

$$\delta(x) = (G - I)\varepsilon(x),$$

so minimising a quadratic form in $\delta(x)$ minimises some other quadratic form in $\varepsilon(x)$.

The most general polynomial acceleration of $w^{(n)}$, also referred to as a semi-iterative technique, is to take linear combinations

$$u^{(n)} = \sum_{i=0}^n \alpha_{n,i} w^{(i)} \quad (3)$$

for $n = 0, 1, \dots$, where

$$\sum_{i=0}^n \alpha_{n,i} = 1. \quad (4)$$

One choice of coefficients $\alpha_{n,i}$, $i = 0, 1, \dots, n$, is to ask that for some symmetric positive definite matrix B the quadratic form

$$\delta(u^{(n)})^T B \delta(u^{(n)}) \quad (5)$$

be minimised.

The conjugate gradient method and its generalisations provide computational procedures for doing this, generally requiring that B has a particular relationship to A and G . Ordinary conjugate gradient acceleration can only be applied to symmetrisable procedures. The acceleration methods ORTHOMIN(∞), ORTHORES(∞) and ORTHODIR(∞) can be applied to non-symmetrisable procedures (see [9] or [4]). The generalised conjugate residual method (see [2]) provides a computational algorithm for minimising the quadratic form (5), when B is the identity matrix, which can be used when A is not symmetric.

The alternative suggested in this paper is to use a matrix B which is diagonal and of less than full rank and to minimise the quadratic form (5) explicitly, as

outlined in the next section. The reduction in the rank of B will reduce the cost of computing the $\alpha_{n,i}$ and numerical experiments suggest that it may have little effect on the speed of convergence under some circumstances.

The novel feature of the technique proposed is that it uses calculations based on a subset of the components of the pseudoresiduals as the basis for acceleration. Chatelin and Miranker [1] provided some support for this approach. They advocate a low-dimensional aggregated version of an equation to help accelerate convergence for an original high dimensional problem.

Sections 2 to 6 of this paper discuss the technique of acceleration by explicit minimisation of the quadratic form (5) when B is of full rank. The difficulties and good features of the technique when B is of full rank need to be investigated as a step towards investigating the properties of the acceleration technique when B is of less than full rank. We shall discuss two problems related to rounding errors, the difficulties of predicting speed of convergence and of testing for convergence, and storage requirements.

The last sections of this paper will discuss some numerical experimentation and practical applications when B is of small rank.

2. Minimisation of the quadratic form

An algorithm for minimising the quadratic form (5) explicitly may be understood by noting that (5) is equal to a quadratic form in $\alpha_{n,i}$ for $i = 0, 1, \dots, n$. Since $\delta(x) = (G - I)x + k$,

$$\begin{aligned} \delta(u^{(n)}) &= (G - I)u^{(n)} + k \\ &= \sum_{i=0}^n (G - I)\alpha_{n,i}w^{(i)} + \sum_{i=0}^n \alpha_{n,i}k \\ &= \sum_{i=0}^n \alpha_{n,i}\delta(w^{(i)}). \end{aligned} \quad (6)$$

Hence

$$\delta(u^{(n)})^T B \delta(u^{(n)}) = \sum_{i=0}^n \sum_{j=0}^n \alpha_{n,i} \alpha_{n,j} \delta(w^{(i)})^T B \delta(w^{(j)}). \quad (7)$$

This quantity may be minimised subject to restriction (4) by the method of Lagrange multipliers. The best values for $\alpha_{n,i}$ for $i = 0, \dots, n$ satisfy the following equations for some real λ :

$$\sum_{j=0}^n \alpha_{n,j} \delta(w^{(i)})^T B \delta(w^{(j)}) - \lambda = 0 \quad \text{for } i = 0, 1, \dots, n \quad (8)$$

and

$$\sum_{i=0}^n \alpha_{n,i} = 1. \tag{9}$$

These equations may be solved by any convenient method to find the $\alpha_{n,i}$ for $i = 0, 1, \dots, n$. An algorithm which is reasonably efficient in both time and storage is Choleski decomposition of the symmetric positive semidefinite submatrix of coefficients of the form $\delta(w^{(i)})^T B \delta(w^{(j)})$ and adding the row and column of extra elements required to form an LU decomposition. Most of the computational effort of decomposition will not need to be repeated for subsequent n .

An example should help to explain some of these ideas. Consider solving the equation

$$\begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{bmatrix} x = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

by an accelerated form of the Jacobi method starting from $w^{(0)} = (1, 0, 0, 0)^T$ and using the identity matrix for B . It is easy to calculate $w^{(1)} = (0, -0.5, 0, 0)^T$, $w^{(2)} = (0.25, 0, 0.25, 0)^T$, $\delta(w^{(0)}) = (-1, -0.5, 0, 0)^T$, $\delta(w^{(1)}) = (0.25, 0.5, 0.25, 0)^T$, $\delta(w^{(0)})^T B \delta(w^{(0)}) = 1.25$, $\delta(w^{(0)})^T B \delta(w^{(1)}) = -0.5$ and $\delta(w^{(1)})^T B \delta(w^{(1)}) = 0.375$.

Now (8) and (9) become

$$\begin{bmatrix} 1.25 & -0.5 & -1 \\ -0.5 & 0.375 & -1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha_{1,0} \\ \alpha_{1,1} \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

and the intended LU decomposition of the matrix of coefficients in this system is

$$\begin{bmatrix} \sqrt{1.25} & 0 & 0 \\ -\sqrt{0.2} & \sqrt{0.175} & 0 \\ \sqrt{0.8} & \sqrt{11.2} & \sqrt{12} \end{bmatrix} \begin{bmatrix} \sqrt{1.25} & -\sqrt{0.2} & -\sqrt{0.8} \\ 0 & \sqrt{0.175} & -\sqrt{11.2} \\ 0 & 0 & -\sqrt{12} \end{bmatrix}.$$

Using this decomposition, it is easy to calculate that $\alpha_{1,0} = 1/3$, $\alpha_{1,1} = 2/3$ and $\lambda = -1/12$. Therefore the linear combination $u^{(1)}$ of $w^{(0)}$ and $w^{(1)}$ such that $\delta(u^{(1)})^T B \delta(u^{(1)})$ is minimised is $(1/3)w^{(0)} + (2/3)w^{(1)} = (1/3, -1/3, 0, 0)^T$. Note also that

$$\delta(u^{(1)}) = (1/3)\delta(w^{(0)}) + (2/3)\delta(w^{(1)}) = (-1/6, 1/6, 1/6, 0)^T$$

and that the result of applying the Jacobi method starting from $u^{(1)}$ is

$$u^{(1)} + \delta(u^{(1)}) = (1/6, -1/6, 1/6, 0)^T = (1/3)w^{(1)} + (2/3)w^{(2)}.$$

This is a particular case of the general result

$$u^{(n)} + \delta(u^{(n)}) = \sum_{i=0}^n \alpha_{n,i} \{w^{(i)} + \delta(w^{(i)})\} = \sum_{i=0}^n \alpha_{n,i} w^{(i+1)}.$$

When solving systems of equations with the same matrix A but different right-hand sides, use can be made of pseudoresiduals calculated for right-hand sides other than the one currently being considered, provided that G does not depend on the right-hand side. If $\delta(x)$ and $\delta(y)$ are for the same right hand side then

$$\delta(x) - \delta(y) = (G - I)(x - y)$$

and so is independent of right hand side. Within the space of vectors of the form

$$u^{(n)} = \sum_{i=0}^n \alpha_{n,i} a^{(i)}$$

subject to

$$\sum_{i \in S_j} \alpha_{n,i} = 0,$$

where S_j is the set of i such that $\delta(a^{(i)})$ has been calculated for the j th right hand side other than the current one, and

$$\sum_{i=0}^n \alpha_{n,i} = 1,$$

we know that

$$\delta(u^{(n)}) = \sum_{i=0}^n \alpha_{n,i} \delta(a^{(i)}),$$

as in (6), and could minimise a quadratic form in $\delta(a^{(n)})$ by the method of Lagrange multipliers. This extension of the method was used for the first of the applications discussed in Section 9 but was not considered for the second application due to limitation of disk space on the computer being used.

3. Rounding errors and truncations of the procedure

There are two problems with the procedure as outlined in the previous section, which are related to rounding errors.

The first problem is that the symmetric matrix containing quadratic forms $\delta(w^{(i)})^T B \delta(w^{(j)})$ may fail to be positive definite because of rounding errors.

Even if rounding errors are not expected to cause any problems, it is recommended that the maximum likely rounding error in $\delta(w^{(i)})^T B \delta(w^{(i)})$ be estimated and that $E_i \alpha_{n,i}$ be added to (8) for $i = 0, 1, \dots, n$, where E_i denotes an estimate of the rounding error in $\delta(w^{(i)})^T B \delta(w^{(i)})$. The modified set of equations will then find $\alpha_{n,i}$ which minimise

$$\delta(u^{(n)})^T B \delta(u^{(n)}) + \sum_{i=0}^n \alpha_{n,i}^2 E_i.$$

This is a reasonable approximation to minimisation of $\delta(u^{(n)})^T B \delta(u^{(n)})$ with likely rounding error in $u^{(n)}$ being considered. The modified set of equations only differs substantially from the unmodified set when the unmodified set is ill-conditioned. The modified set of equations are generally well-behaved.

Computationally, B has always been taken to be diagonal, with ones and zeros on the diagonal. In such cases, $\delta(w^{(i)})^T B \delta(w^{(i)})$ is of the form $\sum_j (Z_j - y_j)^2$ where y_j and Z_j are the components of $w^{(i)}$ and $Gw^{(i)} + k$, respectively, which correspond to the j th one on the diagonal of B . Taking y_j as given and assuming that the rounding error in Z_j is $\varepsilon|Z_j|$ for some small positive constant ε , the rounding error in $(Z_j - y_j)^2$ is approximately $2\varepsilon|Z_j(Z_j - y_j)|$. Making the conservative (and sometimes realistic) assumption that rounding errors add, the rounding error in $\sum_j (Z_j - y_j)^2$ will generally be less than $2\varepsilon \sum_j |Z_j(Z_j - y_j)|$, which was used for E_i in the numerical experiments and applications reported later in the paper.

The second problem is that the numbers $\alpha_{n,i}$ in the linear combinations often become large as n increases. Since $\sum_{i=0}^n \alpha_{n,i} = 1$, large $\alpha_{n,i}$ implies that use of equation (3) would involve subtracting numbers which are much larger than the differences of interest. A major improvement can be sought by noting that all linear combinations of the $w^{(i)}$ are equivalent to linear combinations of the $u^{(i)}$ provided that $\alpha_{n,n}$ is non-zero for all n . For the moment, this assumption will be made. Consequences of its possible failure will be discussed later.

The most general form of modification suggested is to define $v^{(0)} = w^{(0)}$ and, for $n > 0$, to define

$$v^{(n)} = u^{(n-1)} + \delta(u^{(n-1)})$$

if acceleration is performed, at the n th step, but

$$v^{(n)} = v^{(n-1)} + \delta(v^{(n-1)}) = Gv^{(n-1)} + k$$

if acceleration is not performed at the n th step. Provided that $\alpha_{n,n}$ is non zero for all n , all linear combinations of the $w^{(i)}$ may be expressed as linear combinations of the $v^{(i)}$ and the minimisation carried out by calculating quantities of the form $\delta(v^{(i)})^T B \delta(v^{(i)})$.

When acceleration is performed, $u^{(n-1)} + \delta(u^{(n-1)})$ may be computed as a linear combination of known vectors of the form $v^{(i)} + \delta(v^{(i)})$. This is generally cheaper but less precise than computing $v^{(n)}$ directly by applying the basic method starting at $u^{(n-1)}$. For the acceleration strategies below the loss of precision is seldom important, but it is advisable to check the size of the coefficients in such linear combinations and to use the more precise method if these coefficients are large.

This general description requires a prescription as to when acceleration will be performed in order to specify it completely.

In the absence of rounding errors, there would be no point in performing the acceleration until convergence had been achieved to the desired accuracy. Components of $\delta(u^{(n)})$ required for a convergence test could be found using (6). Alternatively, (7) could be used to compute $\delta(u^{(n)})^T B \delta(u^{(n)})$ and a convergence test based on this.

It will generally be desirable to perform the acceleration many times in an attempt to reduce rounding errors by reducing the sizes of coefficients $\beta_{n,i}$ in the relationships

$$u^{(n)} = \sum_{i=0}^n \beta_{n,i} v^{(i)}.$$

Performing acceleration has been found in practice to reduce the average absolute value of $\beta_{n,i}$ for subsequent iterations. This seems intuitively reasonable because acceleration makes the $v^{(i)}$ better approximations to $u^{(n)}$, but counterexamples could probably be constructed.

Ideally, the rule as to when to perform the acceleration ought to depend on the sizes of the coefficients $\beta_{n,i}$ which have been calculated during recent iterations. Such rules have not been investigated because it seemed more important to consider possible ways of truncating the acceleration procedure, and this will influence the choice of when to perform acceleration.

Four modes of acceleration have been considered.

1. Perform the acceleration only once when $\delta(u^{(n)})^T B \delta(u^{(n)})$ will be sufficiently small. This means that the $\alpha_{n,i}$ must be calculated and (8) used after each iteration of the basic method, but that a linear combination of the $w^{(i)}$ is only computed once. This mode was used in one dairying application.

2. The "cheap method of order s " is to perform the acceleration once every $s + 1$ iterations of the basic method. This mode was used in another dairying application.

3. The "expensive method of order s " is to perform the acceleration after every iteration of the basic method but to include only the most recent $s + 1$ of the $v^{(i)}$ in any linear combination. This type of truncation is similar to that of the generalised conjugate gradient acceleration procedures ORTHOMIN, ORTHODIR and ORTHORES.

4. The "intermediate method of order s " is to perform the acceleration after every iteration of the basic method until $u^{(s+1)}$ has been obtained. The procedure is then restarted from this point.

It is expected that the expensive method of order s will converge more rapidly than the intermediate method of order s , which will converge more rapidly than the cheap method of order s for any order. The effect of order and the comparisons between accelerating only once and other methods are likely to be influenced by rounding error.

It is generally true that early iterations from the basic stationary method are given little weight when accelerated approximations are calculated. A modification of the acceleration technique in which, say, 5 or 10 iterations of the basic stationary method were used to find an initial approximation for the accelerated technique would often slightly reduce computing cost. This has not been investigated thoroughly.

4. Speed of convergence

The work of Jennings [6] on the convergence of the conjugate-gradient method cannot be extended to the non-symmetrisable case in any obvious way, since the eigenvalues of G are not all real. However, a crude result on the speed of convergence can be found. Like other theoretical results for the proposed acceleration technique, this result is only strictly applicable when B is symmetric positive definite.

It is easy to show that

$$\delta(w^{(n)}) = G^n \delta(w^{(0)}). \quad (10)$$

Suppose that $\lambda_1, \lambda_2, \dots, \lambda_{m+1}$ are the (possibly complex) eigenvalues of G which are largest in absolute value, and that for $i = 1, 2, \dots, m+1$, p_i is the maximum grade of principal vectors with eigenvalue λ_i . Suppose also that the basic stationary method is convergent so that all eigenvalues of G are smaller than unity in absolute value. Then for $k > 0$,

$$x^{(n)} = G^k \prod_{i=1}^m \left\{ \frac{1}{1-\lambda_i} G - \frac{\lambda_i}{1-\lambda_i} I \right\}^{p_i} \delta(w^{(0)})$$

is of the form

$$\sum_{i=0}^n \alpha_{n,i} \delta(w^{(i)}),$$

where

$$\sum_{i=0}^n \alpha_{n,i} = 1$$

and

$$n = k + \sum_{i=1}^m p_i.$$

Now, using H to denote transpose conjugate,

$$(x^{(n)})^H B x^{(n)} \geq \delta(U^{(n)})^T B \delta(u^{(n)}),$$

since the optimality of $u^{(n)}$ does extend to the case of complex polynomial acceleration. The asymptotic behaviour of $(x^{(n)})^H B x^{(n)}$ is not affected by the m

largest eigenvalues of G since $x^{(n)}$ is orthogonal to all of their principal vectors. Hence the next largest eigenvalue dominates asymptotic behaviour, so there is a constant C such that

$$\delta(u^{(n)})^T B \delta(u^{(n)}) \leq C \lambda_{m+1}^n n^{p_{m+1}-1}$$

for $n \geq \sum_{i=1}^m p_i$.

Note that this bound applies also to any truncated version of the acceleration procedure, provided that the best linear combination of at least $1 + \sum_{i=1}^m p_i$ iterates from the stationary procedure is found.

This result does not tell us much, because the constant C is unknown and the eigenvalues are seldom known precisely. However, it suggests that the proposed acceleration method will converge more rapidly than Chebyshev acceleration because the rate of convergence depends on the actual values of extreme eigenvalues, not merely on the specifications for a region containing them.

5. Testing for convergence

In this section, the basic stationary procedure will be assumed to be convergent; so all eigenvalues of G are assumed to be smaller than unity in absolute value. It will also be assumed that the eigenvectors of G include a basis.

Denote the Euclidean norm of a vector x by $|x|$. Since

$$\varepsilon(x) = -(I - G)^{-1} \delta(x),$$

an estimated bound on $|\varepsilon(x)|$ can be found from $|\delta(x)|$ provided that an estimate of the largest eigenvalue of $(I - G)^{-1}$ can be found. Now the eigenvalues of $(I - G)^{-1}$ are of the form $1/(1 - \lambda)$ where λ is an eigenvalue of G . Since

$$\begin{aligned} |1 - \lambda| &\geq 1 - |\lambda|, \\ |1/(1 - \lambda)| &\leq 1/(1 - |\lambda|). \end{aligned}$$

Therefore the largest eigenvalue of $(I - G)^{-1}$ is not larger than $1/(1 - |M(G)|)$ where $M(G)$ is the eigenvalue of G which is largest in absolute value.

Using (10), a crude estimate of $|M(G)|$ is $|\delta(w^{(n)})|/|\delta(w^{(n-1)})|$ and this can be improved by trying to estimate

$$\lim_{n \rightarrow \infty} \frac{|w^{(n)}|}{|w^{(n-1)}|}$$

by Aitken extrapolation or other techniques. This will be unreliable when $\delta(w^{(0)})$ has a very small component in the direction of the eigenvector of G having largest eigenvalue. If several similar systems of equations are being solved and it is reasonable to assume that the $M(G)$ will be similar, then such estimates should be adequate provided that small estimates of $|M(G)|$ are discarded.

6. Estimation of eigenvectors

When solving several similar sets of equations, it may be desirable to estimate the eigenvectors of the matrix G which have largest eigenvalues. Knowing these eigenvectors, minor modifications to the basic iterative procedure might be made in order to improve the speed of convergence for future work.

Two methods of estimating the eigenvectors of G will be outlined. The reason for using them rather than other methods is that terms of the form $\delta(u^{(i)})^T B \delta(u^{(j)})$ need to be computed for other reasons. Both methods provide good estimates of the largest eigenvalues of G as well as estimates of eigenvectors.

The first method was suggested by A. N. Stokes. Let y denote $\delta(w^{(k)})$ for some k , and suppose that the vectors in the matrix

$$Y = [y, Gy, \dots, G^m y]$$

and $Y^T B Y$ have been computed. If y were in the space, L , spanned by eigenvectors corresponding to the m largest eigenvalues of G then the columns of Y would be linearly dependent. So there would be a vector α such that $Y\alpha = 0$, the eigenvalues would be roots of

$$(1, \lambda, \lambda^2, \dots, \lambda^m)\alpha = 0$$

and the eigenvectors could be found as linear combinations of the vectors in Y where the coefficients in the linear combinations are those of powers of λ in polynomials of the form

$$(\lambda - \lambda_1)(\lambda - \lambda_2) \cdots (\lambda - \lambda_m).$$

Note that α would be an eigenvector of the matrix $Y^T B Y$ and would have eigenvalue zero. Provided that y was not in any subspace of L spanned by fewer than all of the eigenvectors, $Y^T B Y$ would have no other eigenvectors with eigenvalue zero.

If y is close to L then the eigenvector corresponding to the smallest eigenvalue of $Y^T B Y$ will be close to α , since eigenvectors of isolated eigenvalues are continuous functions of a matrix. This method was used for estimating the eigenvalues quoted in the final section of this paper. It can only be applied if the mode of acceleration has provided a sequence of approximations during which no acceleration was performed.

A second method is applicable even if acceleration has been performed several times. However it is slightly less accurate than the first method when both are applicable.

In outline, the method consists of using Gram-Schmidt's orthogonalisation to find a basis for $\delta(v^{(m)})$, $\delta(v^{(m+1)})$, \dots , $\delta(v^{(m+n-1)})$; making the approximation that $G\delta(v^{(m+n-1)})$ is equal to its projection onto the space spanned by

$\delta(v^{(m)}), \dots, \delta(v^{(m+n-1)})$; using this approximation and the knowledge that $G\delta(v^{(i)}) = \delta(v^{(i+1)})$ for $i = m, m+1, \dots, m+n-1$ to find an explicit form for G relative to the basis; and finding the eigenvalues of this approximation to G . This is computationally practical for large systems of equations only if attention is restricted to a subset of the components of the vectors. This will often be adequate for the proposal of considering possible modifications to the basic iterative procedure.

7. Use of a matrix B of small rank

The method of acceleration being discussed in this paper is computationally attractive only when the matrix B is taken to be diagonal and of small rank, so that the inner products of the form $\delta(v^{(i)})^T B \delta(v^{(j)})$ are relatively inexpensive to compute.

The matrix B is somewhat arbitrary with this acceleration method, but some guidelines are as follows. Its rank should be substantially larger than the expected number of iterations so that the quantity being minimised is unlikely to vanish before convergence is achieved. It should be diagonal in order to reduce computing cost. It should have non-zeros (usually ones) on the diagonal for components which are expected to converge slowly, differentially from other components or in an unknown fashion. When background knowledge does not help in choosing B , a reasonable strategy might be to have $10t$ randomly placed ones on the diagonal where t is the expected total number of iterations.

When B is of small rank, it is generally recommended that an iteration of the basic iterative method be done using a proposed solution as starting point in order to check that the pseudoresidual is indeed small as suggested by the small value of the quadratic form. It is also recommended that computer programs for solving large systems have the facility to be restarted using a different choice for B and not need to repeat calculations concerning the basic stationary method.

As the impact of using a matrix B of small rank seems difficult to investigate theoretically, it has been investigated using numerical experimentation. Some practical applications will then be discussed.

8. Numerical experimentation

The equation used as the basis for most numerical experiments was the finite-difference analogue of Laplace's equation on a rectangular grid of $29 \times 34 = 986$ points. Denoting the value for the grid point in the i th row and j th column by

$x_{i,j}$, the scalar equations were of the form

$$4x_{i,j} - x_{i-1,j} - x_{i,j-1} - x_{i+1,j} - x_{i,j+1} = 0$$

with terms not belonging to the grid being taken as zero. The solution to these equations is that all variables are zero. The initial approximation used was generated by a pseudo-random number generator with $x_{i,j}$ being independently selected from a uniform distribution over the range $(-0.5, 0.5)$.

It should be remembered that the methods being discussed here are not computationally competitive for solving Laplace's equation. Rather, Laplace's equation is being used as a test problem because it has been thoroughly investigated by other methods. A square grid was not used because the matrices (denoted G) of stationary iterative methods, particularly the Jacobi method, might have repeated eigenvalues and the result of Section 4 suggests that the acceleration technique might perform atypically well when there are repeated eigenvalues. A nearly square grid was used so that the convergence of the methods of Jacobi and Gauss-Seidel is slow.

When the expensive acceleration method was used (with linear combinations being taken whenever possible) and B being the identity matrix, the performance of the acceleration method when Gauss-Seidel was used as the stationary method is indicated in Table 1.

TABLE 1. This table indicates the performance of the *expensive* acceleration method as a function of the order of the method when accelerating Gauss-Seidel's method for Laplace's equation on a 29×34 grid. Figures in parentheses are estimates based on runs which were terminated before the specified precision was reached.

Numbers of iterations until norm of pseudoresidual reached

Order of acceleration	10^{-5}	10^{-10}	10^{-15}
0	319	(1490)	(2695)
1	78	448	(968)
2	64	413	(716)
3	64	134	211
5	64	131	199
10	63	130	192
20	63	127	185
30	63	130	190
40	63	123	178
60	62	104	138
80	62	90	126
100	62	90	117

Information which might aid the interpretation of this and later tables is as follows.

1. The norm (meaning Euclidean norm) of the pseudoresidual for the first iteration was 8.8.

2. Spectral radii given in Table 2 of [7] indicate that for a 31×31 grid the spectral radius of Young's SOR is .797 and that of Chebyshev accelerated Gauss-Seidel is .857. This suggests that these techniques would take about 102 and 149 iterations, respectively, to reduce the norm of the pseudoresidual by a factor of 10^{10} .

3. The optimum value for the parameter ω of SOR is near 1.82. The norm of the pseudoresidual was found to reach 10^{-5} , 10^{-10} and 10^{-15} after 85, 145 and 211 iterations for $\omega = 1.82$.

4. Computations were done to a precision of about 16 decimal significant figures.

TABLE 2. This table indicates the performance of the *cheap* acceleration method as a function of the order of the method when accelerating Gauss-Seidel's method for Laplace's equation on a 29×34 grid. Figures in parentheses are estimates based on runs which were terminated before the specified precision was reached.

Numbers of iterations until norm of pseudoresidual reached

Order of acceleration	10^{-5}	10^{-10}	10^{-15}
0	319	(1490)	(2695)
1	86	229	(636)
2	90	222	411
3	84	220	347
5	78	174	271
10	67	165	253
20	80	170	290
30	93	193	310
40	82	205	311
60	122	244	366
80	157	243	369
100	106	303	404

Table 2 shows the performance of the *cheap* acceleration method as a function of the order of the method. The reason why performance drops off with increasing order after about order 10 seems to be that rounding errors restrict the improvement made by taking linear combinations once the weights in the linear combination reach about 10^5 or 10^6 , and linear combinations are taken less often as order increases. The pattern of change of performance with order is somewhat erratic because the linear combinations are taken after numbers of iterations which vary with the order.

For the intermediate acceleration method, the norm of the pseudoresidual was found to reach 10^{-5} , 10^{-10} and 10^{-15} after 75, 164 and 270 iterations for order 5 and after 65, 155 and 243 iterations for order 20. The improvement in

convergence with increasing order supports the explanation given in the previous paragraph.

Tables 3, 4 and 5 show the results of some numerical experiments investigating the effect of the rank of B on rate of convergence.

TABLE 3. This table indicates the performance of the *expensive* acceleration method of order 10 as a function of the rank of the matrix B when B was taken to be diagonal with ones and zeros on the diagonal. The location of non-zero elements on the diagonal of B was determined using a pseudorandom number generator. For the rank of B being 100 and 300, three different choices for B were tried.

Rank of B	Number of iterations until norm of pseudoresidual reached		
	10^{-5}	10^{-10}	10^{-15}
986	63	130	192
300	65	129	196
	66	125	191
	65	128	182
100	66	131	203
	70	129	204
	68	136	193
	71	141	209
50	89	191	342
30	112	251	347
20	148	317	414
15	78	103	134
10			

For the expensive acceleration method of order 10, Table 3 gives results which suggest that the rate of convergence decreases substantially once the rank of B becomes less than the number of iterations taken. Such behaviour is to be expected because the minimising procedure becomes more capable of reducing the quadratic form $\delta(u^{(n)})^T B \delta(u^{(n)})$ in ways which do not reduce the norm of $\delta(u^{(n)})$ in these circumstances.

The remarkably good performance of the expensive acceleration procedure of order 10 when B was of rank 10 deserves special attention. Similarly good performance was found when different matrices of rank 10 were used for B , when the constant ε used in rounding error computations was changed slightly, when the numerical experiment was repeated on a different computer, and when a matrix of rank 5 was used for expensive acceleration of order 5. However, in some of these cases convergence was only satisfactory until the norm of the pseudoresidual reached about 10^{-13} , and when a matrix of rank 3 was used for expensive acceleration of order 3, convergence was unsatisfactory.

During early iteration, the procedure finds a linear combination $u^{(n)}$ which reduces $\delta(u^{(n)})^T B \delta(u^{(n)})$ to as small a value as could be expected given the magnitudes of rounding errors. This can be seen as the dashed line in Figure 1. It

takes until about iteration 80 before there is any further substantial reduction in $\delta(u^{(n)})^T B \delta(u^{(n)})$ yet over this period the acceleration technique has been effective as judged by reductions by a factor of about 10^{-5} in the usual three measures of convergence, the norms of the pseudoresidual, of the residual and of the error. The rate of convergence seems to increase from that point, unlike the convergence of most other techniques which falls off as the number of iterations increases. The convergence seems erratic but the phenomenon might be worth investigating further.

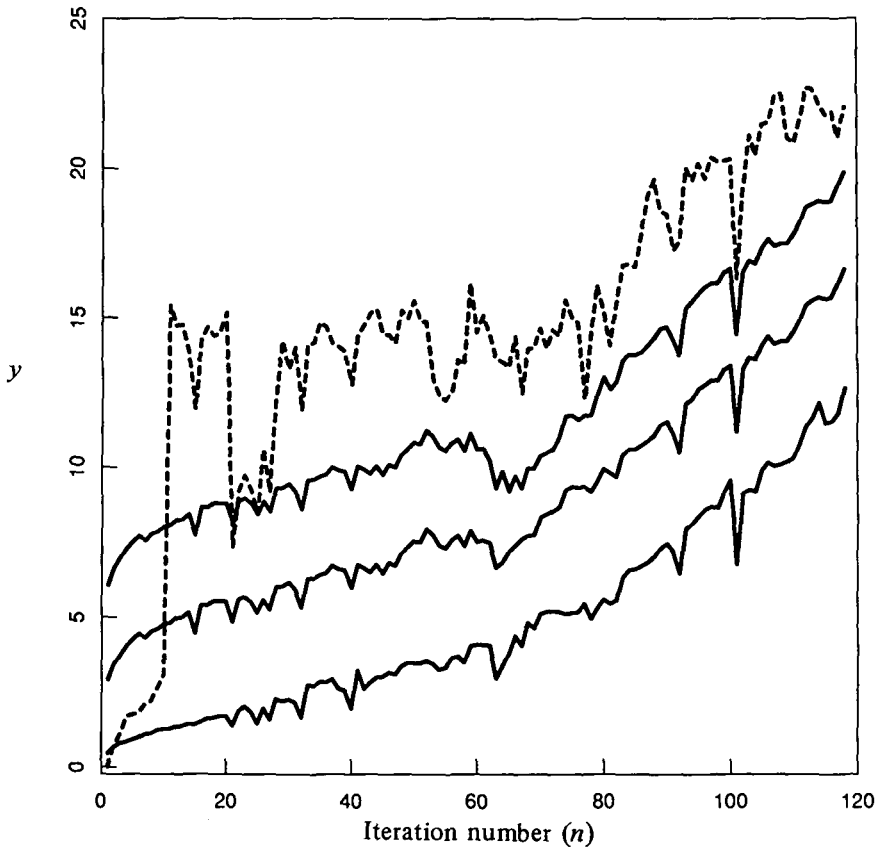


FIGURE 1. Graphs showing the performance of the expensive acceleration procedure of order 10 when B was of rank 10. The curves from top to bottom are

$$\begin{aligned}
 y &= \log_{10}\{\delta(u^{(n)})^T B \delta(u^{(n)})\}, \\
 y &= \log_{10}\{|\delta(u^{(n)})|\} + 7, \\
 y &= \log_{10}\{|Au^{(n)} - b|\} + 2, \text{ and} \\
 y &= \log_{10}\{|u^{(n)} - \bar{u}|\} + 1.
 \end{aligned}$$

Note that the trends of the graphs in Figures 1 and 2 may be interpreted as indicating how the number of significant figures of precision, as judged by various criteria, increases with iteration number.

For the expensive acceleration method of order 100, Table 4 shows that reducing the rank of B has more influence than for order 10. The procedure is more capable of reducing $\delta(u^{(n)})^T B \delta(u^{(n)})$ in ways which do not reduce the norm of $\delta(u^{(n)})$ than is order 10 acceleration for B of a given rank. The performance of expensive acceleration of order 100 when B is of rank 100 seemed unremarkable. It is possible that larger numbers of iterations and possibly reduced rounding errors would be required before rapid convergence would occur like that for order 10 when B was of rank 10, even if it were generally true that making the rank of B equal to the order always gave particularly rapid convergence.

TABLE 4. This table indicates the performance of the *expensive* acceleration technique of order 100 as a function of the rank of the matrix B , when B was taken to be diagonal with ones and zeros on the diagonal. The location of non-zero elements on the diagonal of B was determined using a pseudorandom number generator.

Rank of B	Number of iterations until norm of pseudoresidual reached		
	10^{-5}	10^{-10}	10^{-15}
986	62	90	117
300	96	113	129
150	108	126	143
100	103	119	182

For the cheap acceleration method, Table 5 indicates that there is not a substantial decrease in rate of convergence until the rank of B becomes nearly as small as the order of acceleration. Presumably, this is because the cheap acceleration method is much less capable of reducing the ratio of $\delta(u^{(n)})^T B \delta(u^{(n)})$ to the square of the norm of $\delta(u^{(n)})$ than the expensive acceleration method because the vectors involved in successive linear combination phases of the computation are distinct, and so reductions in this ratio which often occur at linear combination steps do not accumulate to as great an extent.

TABLE 5. This table indicates the performance of the *cheap* acceleration method of order 10 as a function of the rank of the matrix B , when B was taken to be diagonal with ones and zeros on the diagonal. The location of non-zero elements on the diagonal of B was determined using a pseudorandom number generator.

Rank of B	Number of iterations until norm of pseudo-residual reached		
	10^{-5}	10^{-10}	10^{-15}
986	67	165	253
300	68	179	265
100	77	173	257
50	78	181	280
30	78	173	271
20	78	171	260
15	82	185	262
10	163	361	472

Figure 2 shows the convergence of cheap acceleration of order 10 for B of rank 100 in detail. The taking of linear combinations often actually increases the norms of the pseudo-residual and of the residual but they are then rapidly decreased by the Gauss-Seidel steps. The convergence of expensive acceleration is generally smoother than the convergence of cheap acceleration, and so a figure illustrating the convergence of expensive acceleration methods has not been included here.

One objection which has been raised to this acceleration technique is that the choice of B is arbitrary. Table 3 indicates the performance of the technique for three different choices of B for each of the ranks 100 and 300. The same pseudorandom number generator was used for generating the positions of the ones on the diagonal of B in all cases, but different seeds were used. The average coefficient of variation of the number of iterations required to reach a given precision is about 3 percent based on these results. This suggests that the arbitrariness of B is not a major problem.

Acceleration of SOR showed some promise despite the result of Varga [8] suggesting that acceleration of SOR would not do better than SOR itself. See Table 6. This indicates that the best value for ω when SOR is accelerated is less than when it is not accelerated. However, like all the results of this section, this should only be regarded as an indicative result since it is for only one equation.

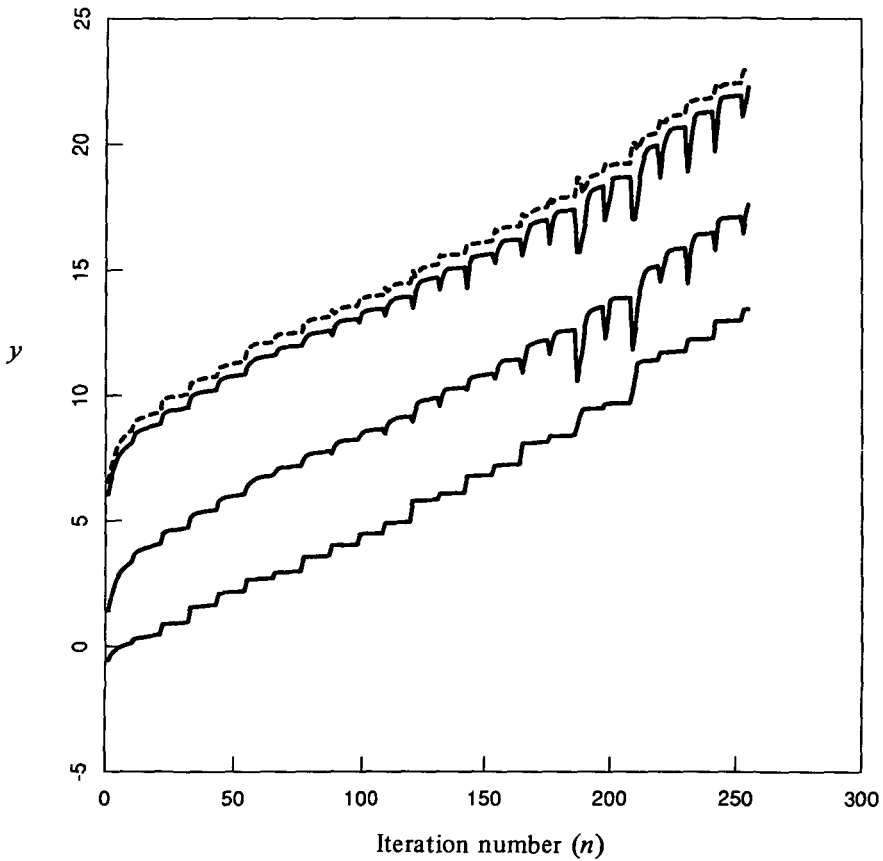


FIGURE 2. Graphs showing the performance of the cheap acceleration procedure of order 10 when B was of rank 100. The curves from top to bottom are

$$y = \log_{10}\{\delta(u^{(n)})^T B \delta(u^{(n)})\} + 7,$$

$$y = \log_{10}\{|\delta(u^{(n)})|\} + 7,$$

$$y = \log_{10}\{|Au^{(n)} - b|\} + 2.5, \text{ and}$$

$$y = \log_{10}\{|u^{(n)} - \bar{u}|\}.$$

TABLE 6. This table compares the performance of SOR with accelerated forms of SOR.

Method	Number of iterations until norm of pseudoresidual reached		
	10^{-5}	10^{-10}	10^{15}
Unaccelerated SOR with $\omega = 1.81$	82	164	251
Best unaccelerated SOR ($\omega = 1.82$)	85	145	211
Unaccelerated SOR with $\omega = 1.83$	90	145	211
Cheap acceleration of order 10 applied to SOR with $\omega = 1.60$	78	147	201
Cheap acceleration of order 10 applied to SOR with $\omega = 1.76$	75	134	194
Cheap acceleration of order 10 applied to SOR with $\omega = 1.82$	123	227	302

Two additional numerical experiments were done using the Jacobi method as the stationary iterative technique to solve Laplace's equation on a grid of $9 \times 9 = 81$ points. This example was chosen because it is sufficiently simple for examples of undesirable behaviour to be readily constructed yet not so small as to be unrealistic.

The expensive acceleration technique of order 5 was used with three different sets of components being used in the inner products. When the components used were the middle row of the grid, 91 iterations were required to reach a preassigned precision. When the first row of the grid was used 76 iterations were required. For a randomly chosen set of components, 68 iterations were required. The poor performance obtained by using the middle row of the grid is due to the fact that some eigenvectors of the iteration matrix for this process vanish at all of these points, so no acceleration of these eigenvectors can be expected.

Possible breakdown of the technique was investigated by using an initial approximation such that

$$x_{i,j} = 1 \quad \text{for } 1 < i < 9 \quad \text{and} \quad 1 < j < 9$$

and using components $x_{i,j}$ for $2 < i < 8$, $2 < j < 8$ for the inner products. Since $w^{(0)}$ and $w^{(1)}$ do not differ on components used for the inner products, $\delta(w^{(0)})^T B \delta(w^{(0)}) = 0$. Hence any minimisation which may take $w^{(0)}$ as the best linear combination will do so (neglecting the rounding error modifications). Both cheap and expensive acceleration methods of order 5 converged satisfactorily, although no real progress towards convergence was made until the smallness of the order stopped $w^{(0)}$ from being included in linear combinations.

The algorithm could be said to have broken down, but the iterative process was continued because convergence checks based on all components of pseudo-residuals showed that convergence had not been achieved despite the small values of $\delta(u^{(n)})^T B \delta(u^{(n)})$. Real convergence was eventually achieved.

Similar break-down with a satisfactory result occurred for the acceleration of order 10 when B was of rank 10 reported in Table 3.

In Section 3 it was assumed that $\alpha_{n,n}$ was non-zero for all n . These break-downs of the procedure are associated with the failure of this assumption. It appears that the possible failure of the assumption is not a problem provided that the process is not terminated before real convergence is achieved.

9. Applications

The acceleration method was developed in 1980–81 while the author was employed by the Australian Dairy Herd Improvement Scheme supported by the Australian Dairy Research Committee. The method was peripheral to the objectives of the scheme, and so the properties of the method were only investigated sufficiently for the method to be applied to two particular classes of simultaneous equations. For both classes, the coefficient matrix A was symmetric positive definite and Gauss-Seidel was used as the basic stationary method since preliminary testing suggested that the best accelerated successive over-relaxation technique had parameter ω near unity. The statistics collected on the performance of the acceleration method were less extensive than now seems desirable but should be of some interest.

The first class of simultaneous equations involved one equation for each bull of a breed. The method of forming the equations followed [3] very closely. The largest such set of equations involved 14161 bulls of Friesian breed. There were 409743 non-zero elements in the coefficient matrix A with non-zeros occurring almost whenever two bulls had daughters or daughters of daughters which had first lactations influenced by the same environmental effect. It is estimated that the largest eigenvalues of G were 0.74 and 0.66.

Rounding error was not a problem, and so a linear combination of vectors with 14161 components was only calculated once. The matrix B had rank 447 and was chosen to include many of the most widely-used bulls plus a few lesser-used bulls from each Australian state.

Convergence was satisfactory, 11 iterations being required to reduce the pseudo-residual by a factor of about 10^4 . However the work required to implement the acceleration procedure would not have been justifiable except that experience in the use of the method had been obtained. Many other procedures would have been satisfactory, including unaccelerated Gauss-Seidel which would have

required only about 20 iterations for satisfactory precision to be obtained. Points to note are the following.

1. The choice of rank of B caused little problem. It was desirable that it be much less than the number of equations (14161) in order to reduce computing costs and that it be larger than the number of iterations. The number of iterations was expected to be less than 30 given other people's experience.

2. The properties of the basic iterative method depend upon the pattern of farmers' choices of mates for their cows and on the decisions of bull-breeding organisations. Slight variations of genetic model were also tried. For these reasons, non-adaptive Chebyshev acceleration and SOR seemed unlikely to be reliable.

3. Equations were solved for four right-hand-side vectors, corresponding to two different standardisations of each of milk yield and fat yield. Using all available pseudoresiduals from previous right-hand sides, as discussed near the end of Section 2, the second, third and fourth right-hand sides required 9, 7 and 6 iterations, respectively, for the required precision to be achieved.

The second class of simultaneous equations was set up essentially as described in [5] with one equation per bull and two per cow, plus some other equations. The largest such set of equations was also for the Friesian breed and involved slightly more bulls than for the earlier technique because more data had been obtained. The total number of simultaneous equations to be solved was about 970,000. This was computationally practicable on a computer with about 500,000 words of memory, only because most of the calculations could be handled one herd at a time for each iteration.

It is estimated that the two largest eigenvalues of G were 0.94 and 0.84. Typically, about 30 iterations have been required for reducing the pseudoresidual by a factor of 10^4 with sets of equations like this, and more than 80 percent of the computing time for solving the simultaneous equations has been expended on the basic Gauss-Seidel computations.

Rounding error appeared to be important. The cheap acceleration method of order 10 was used. The matrix B had rank 6007 which was probably larger than necessary but not so large that computation of terms of the form $\delta(v^{(i)})^T B \delta(v^{(j)})$ became a major component of computing cost.

The two points made about the first class of simultaneous equations are applicable here also, except that there was no information available on the convergence of similar sets of equations.

For a very large system of equations like this, the previous approximations are necessarily held on disk or some other form of backing store. In this application, storage limitations had little influence on the choice of acceleration algorithm. However, computational cost of doing operations on vectors was substantial, and this was the reason for choosing the cheap form of acceleration.

As with the first class of simultaneous equations, the acceleration technique was probably not optimal. However it performed satisfactorily and did not require any advance knowledge except an inkling of an idea about the order of the number of iterations which would be required.

10. Discussion

The acceleration technique discussed in this paper is not well-defined. More research and experimentation are required in order to provide further guidelines as to when and how it should be used.

The generalised conjugate-gradient procedures ORTHOMIN, ORTHODIR and ORTHORES are the most obvious alternatives to the proposed technique. When the matrix B is of full rank, the proposed technique is of similar computational cost to the alternatives, whether or not all procedures are truncated. The speed of convergence should also be similar, at least for nontruncated procedures, since all of the techniques are Krylov subspace methods which are optimal relative to some norm.

Tables 3, 4 and 5 indicate that the speed of convergence does not decrease very greatly as the rank of B is decreased. One component of the computational expense of applying the acceleration is that of computing inner products. This is approximately proportional to the rank of B times the number of iterations for truncated acceleration techniques. This component of computational expense is generally much greater when B is of full rank than when the rank of B is reduced. The computational expenses of the basic iterative technique and taking linear combinations of approximations need also be considered, but these tables do suggest that overall computing costs would often be reduced by choosing B to be of substantially less than full rank. Since the proposed technique with B of full rank is thought to have similar performance and cost to ORTHOMIN, ORTHODIR and ORTHORES, this suggests that the proposed technique will often be better than those alternatives.

The numerical experimentation conducted suggests that the expensive acceleration method of low order, with the rank of B being equal to the order, is particularly effective. This may not be generally true, but it seems worthy of some investigation. Apart from this, the expensive acceleration of order 3 to 5 and the cheap acceleration of order about 10 seem to be most likely to be worthwhile. More complex rules as to when to take linear combinations of previous iterations might also be investigated.

The feature of the proposed method, that it enables eigenvectors of the basic iterative technique to be estimated, is most likely to be advantageous in situations where little is known about the equations being solved. Knowledge about

such eigenvalues might help by suggesting modifications to the basic iterative technique.

The technique is most likely to be competitive with other techniques when the number of linear equations is very large, so that B may be chosen to be of very much less than full rank, and when little is known about the properties of the system of equations, so that specialised methods are less likely to be applicable.

References

- [1] F. Chatelin and W. L. Miranker, "Acceleration by aggregation of successive approximation methods", *Linear Algebra Appl.* **43** (1982) 17–47.
- [2] S. C. Eisenstat, H. C. Elman and M. H. Schultz, "Variational iterative methods for nonsymmetric systems of linear equations", *SIAM J. Numer. Anal.* **20** (1983) 345–357.
- [3] R. W. Everett, R. L. Quaas and A. C. McClintock, "Dairy sire evaluation considering genetic merit of daughter's maternal grandsire", *J. Dairy Sci.* **62** (1979) 1304–1313.
- [4] L. A. Hageman and D. M. Young, *Applied Iterative Methods* (Academic Press, 1981).
- [5] C. R. Henderson, "Use of all relatives in intraherd prediction of breeding values and producing abilities", *J. Dairy Sci.* **58** (1975) 1910–1916.
- [6] A. Jennings, "Influence of the eigenvalue spectrum on the convergence rate of the conjugate gradient method", *J. Inst. Maths. Applics.* **20** (1977) 61–72.
- [7] D. M. Martin and G. J. Tee, "Iterative methods for linear equations with symmetric positive definite matrix", *Comput. J.* **4** (1961) 242–254.
- [8] R. S. Varga, "A comparison of the successive over relaxation method and semi-iterative methods using Chebyshev polynomials", *J. Soc. Indus. Appl. Math.* **5** (1957) 39–46.
- [9] D. M. Young and K. C. Jea, "Generalised conjugate gradient acceleration of non-symmetrisable iterative methods", *Linear Algebra Appl.* **34** (1980) 159–194.