SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

© 1985 Society for Industrial and Applied Mathematics

010

MULTI-SPLITTINGS OF MATRICES AND PARALLEL SOLUTION OF LINEAR SYSTEMS*

DIANNE P. O'LEARY† AND R. E. WHITE‡

Abstract. We present two classes of matrix splittings and give applications to the parallel iterative solution of systems of linear equations. These splittings generalize regular splittings and P-regular splittings, resulting in algorithms which can be implemented efficiently on parallel computing systems. Convergence is established, rate of convergence is discussed, and numerical examples are given.

Key words. matrix splittings, iterative methods for linear systems, parallel computation, regular splittings

AMS(MOS) subject classifications. 65F10, 65N20

1. Introduction. Consider the solution of a large linear system of equations

$$Ax=b$$

on a parallel computer. We assume that several processors are available and that they can execute different instruction sequences on their local data and can communicate with physically adjacent processors.

In this paper we consider the problem of solving linear systems for which the matrix A can either be split into many pieces or split into two pieces in many ways. An example of the first case is the assembly of a finite element matrix by elements. In that case A can be decomposed as

$$A=\sum_{k=1}^{K} A_k$$

where each matrix A_k has small rank. The second case arises from having several candidate iterative methods

$$B_k x_{i+1}=C_k x_i+b, \qquad i=0, 1, \cdots,$$

where for $k=1, 2, \cdots, K$, $A=B_k-C_k$.

We discuss ways of using these two kinds of decompositions of A in order to construct convergent iterative methods which are structured so that most operations can be performed in parallel. We base such iterative methods on *multi-splittings* of the matrix A.

In § 2 we define multi-splittings and prove some convergence results for these iterative methods. Section 3 provides a discussion of parallelism in the iterative methods, examples of problems for which multi-splittings can be used, and motivation for the definitions and results of § 2. Section 4 provides results of some numerical experiments on multi-splittings. It is possible to read §§ 3 and 4 before § 2 if a reader is so inclined.

2. Multi-splittings: definitions and theory. We begin with a definition of a multi-splitting of a matrix A, discuss its use in an iterative method for solving linear systems, and prove some convergence results. For notational convenience we omit the lower and upper limits 1 and K on all sums and the indices $k=1, \cdots, K$ on ordered triples (B_k, C_k, D_k).

* Received by the editors January 24, 1984, and in revised form April 19, 1984.

† Computer Science Department, University of Maryland, College Park, Maryland 20742. The work of this author was supported by the Air Force Office of Scientific Research under grant AFOSR-82-0078.

‡ Department of Mathematics, North Carolina State University, Raleigh, North Carolina 27695-8205.

DEFINITION. Let A, B_k, C_k, and D_k be $n\times n$ matrices. Then (B_k, C_k, D_k) is called a *multi-splitting of* A if

i) $A=B_k-C_k$, $k=1,\cdots,K$, where each B_k is invertible.

ii) $\sum_k D_k=I$ where the matrices D_k are diagonal and $D_k\geqq 0$.

We will use the notation

$$\mathbf{H}=\sum_k D_kB_k^{-1}C_k \quad\text{and}\quad \mathbf{G}=\sum_k D_kB_k^{-1}.$$

We are interested in the convergence of an iterative method based on $\mathbf{H}$ and $\mathbf{G}$ for solving $Ax=b$. Using (i) above, $Ax=b$ may be written as

$$B_kx=C_kx+b, \qquad k=1,\cdots,K$$

or

$$x=B_k^{-1}C_kx+B_k^{-1}b, \qquad k=1,\cdots,K.$$

We use the weighting matrices D_k to combine these K equations as

$$\sum_k D_kx=\sum_k D_kB_k^{-1}C_kx+\sum_k D_kB_k^{-1}b,$$

which, by (ii) and the definitions of $\mathbf{H}$ and $\mathbf{G}$ yields the following algorithm.

```
ALGORITHM 1
Choose $x_0$ arbitrarily.
For $i=0,1,2,\cdots$, until convergence
    $x_{i+1}=\mathbf{H}x_i+\mathbf{G}b$.
```

The parallelism in a variant of this algorithm will be discussed in § 3.

It would be convenient if it were true that whenever the iterative methods based on each of the splittings $A=B_k-C_k$ converged, then Algorithm 1 produced a convergent sequence, too. Unfortunately, the situation is more complicated than that, as the following trivial example shows.

Example. Let $K=2$, $n=2$, and consider

$$A=\begin{bmatrix}\frac{3}{4} & 0\\ 0 & \frac{3}{4}\end{bmatrix}=B_1-C_1=B_2-C_2,$$

where

$$B_1=\begin{bmatrix}.5 & -1\\ 1 & 4\end{bmatrix}, \qquad C_1=\begin{bmatrix}-.25 & -1\\ 1 & 3.25\end{bmatrix},$$

and

$$B_2=\begin{bmatrix}4 & 1\\ -1 & .5\end{bmatrix}, \qquad C_2=\begin{bmatrix}3.25 & 1\\ -1 & -.25\end{bmatrix}.$$

Then

$$B_1^{-1}C_1=\begin{bmatrix}0 & -.25\\ .25 & .875\end{bmatrix}, \qquad B_2^{-1}C_2=\begin{bmatrix}.875 & .25\\ -.25 & 0\end{bmatrix}.$$

The spectral radius ρ for both matrices is .7965, so iterations based on both splittings are convergent. But, with the choice

$$D_1=\begin{bmatrix}0 & 0\\ 0 & 1\end{bmatrix}, \qquad D_2=\begin{bmatrix}1 & 0\\ 0 & 0\end{bmatrix},$$

the resulting iteration matrix is

$$\mathbf{H}=\begin{bmatrix}.875 & .25\\ .25 & .875\end{bmatrix}$$

for which the spectral radius is 1.125, and therefore Algorithm 1 would not be convergent. Other choices of D_1 and D_2 will change this situation, of course. For example, if the definitions of D_1 and D_2 above are interchanged, the resulting matrix $\mathbf{H}$ has spectral radius equal to $\frac{1}{4}$.

Recall that (B, C) is a weak regular splitting of A if $A=B-C$, $B^{-1}\geqq 0$, and $H\equiv B^{-1}C\geqq 0$. Similarly, (B, C) is called a P-regular splitting of A if B is nonsingular and the symmetric part of $B+C$ is positive definite. From standard results in the theory of iterative methods (see, for example, [1] and [6]),

(1) If (B, C) is a weak regular splitting of a matrix A satisfying $A^{-1}\geqq 0$, and $H=B^{-1}C$, then $\rho(H)<1$.

(2) If (B, C) is a P-regular splitting of a symmetric positive definite matrix A, and $H=B^{-1}C$, then $\rho(H)<1$.

(3) If $\|H\|<1$ for any matrix norm, then $\rho(H)<1$.

We seek conditions on the multi-splitting (B_k, C_k, D_k) which will ensure that analogous results apply to the splitting resulting in $\mathbf{H}$. Consequently, these conditions will ensure that Algorithm 1 is convergent. The example above shows that the second result does not have a direct analogue: it is not enough that each splitting in the multisplitting is a P-regular splitting of a symmetric positive definite matrix. The other two results do generalize without additional hypotheses.

THEOREM 1. (a) *If, for $k=1,2,\cdots,K$, (B_k, C_k) is a weak regular splitting of a matrix A satisfying $A^{-1}\geqq 0$, then Algorithm 1 is convergent.*

(b) *If, for $k=1,2,\cdots,K$, (B_k, C_k) is a P-regular splitting of a symmetric positive definite matrix A and $D_k=\alpha_k I$, then Algorithm 1 is convergent.*

(c) *If, for $k=1,2,\cdots,K$, $\|B_k^{-1}C_k\|_\infty<1$, then Algorithm 1 is convergent.*

Proof. (a) The proof parallels the proof for convergence of weak regular splittings found, for example, in Ortega [6]. From the definitions of $\mathbf{H}$ and weak regular splitting we have the following three facts:

1. $\mathbf{H}\geqq 0$ and therefore $\mathbf{H}^j\geqq 0$, $j=0,1,\cdots$.
2. $I-\mathbf{H}=\sum_k D_k B_k^{-1}A$.
3. $(I+\mathbf{H}+\cdots+\mathbf{H}^m)(I-\mathbf{H})=I-\mathbf{H}^{m+1}$.

Now, using these facts in order,

$$\begin{aligned}0&\leqq(I+\mathbf{H}+\cdots+\mathbf{H}^m)\sum_k D_k B_k^{-1}\\ &=(I+\mathbf{H}+\cdots+\mathbf{H}^m)(I-\mathbf{H})A^{-1}\\ &=(I-\mathbf{H}^{m+1})A^{-1}\leqq A^{-1}.\end{aligned}$$

Therefore, the elements of $\mathbf{H}^m$ must remain bounded, and therefore $\mathbf{H}$ is convergent.

(b) Again, the proof parallels a standard proof of convergence, that for P-regular splittings [6]. It is sufficient to show that $A-\mathbf{H}^TA\mathbf{H}$ is positive definite; then the result that $\rho(\mathbf{H})<1$ follows from a theorem of Stein [9]. We use the notation $B^{-T}\equiv(B^T)^{-1}=(B^{-1})^T$.

$$\begin{aligned}
A-\mathbf{H}^TA\mathbf{H}&=A-\Big(I-\sum_k D_kB_k^{-1}A\Big)^TA\Big(I-\sum_k D_kB_k^{-1}A\Big)\\
&=\sum_k AD_kB_k^{-1}A+\sum_k AB_k^{-T}D_kA-\sum_{k,j} AB_k^{-T}D_kAD_jB_j^{-1}A\\
&=\sum_k AB_k^{-T}[B_k^TD_k+D_kB_k-D_kAD_k]B_k^{-1}A\\
&\quad-\sum_{\substack{k,j\\k\neq j}} AB_k^{-T}D_kAD_jB_j^{-1}A\\
&=\sum_k AB_k^{-T}[B_k^TD_k+D_kA+D_kC_k-D_kAD_k]B_k^{-1}A\\
&\quad-\sum_{\substack{k,j\\k\neq j}} AB_k^{-T}D_kAD_jB_j^{-1}A\\
&=\sum_k AB_k^{-T}[B_k^TD_k+D_kC_k+\sum_{\substack{j\\j\neq k}} D_kAD_j]B_k^{-1}A\\
&\quad-\sum_{\substack{k,j\\k\neq j}} AB_k^{-T}D_kAD_jB_j^{-1}A\\
&=\sum_k AB_k^{-T}[B_k^TD_k+D_kC_k]B_k^{-1}A\\
&\quad+\sum_{\substack{k,j\\k\neq j}} AB_k^{-T}D_kAD_j[B_k^{-1}A-B_j^{-1}A]\\
&\equiv \mathbf{S}_1+\mathbf{S}_2.
\end{aligned}$$

Let sym $(P)\equiv(P+P^T)/2$ denote the symmetric part of the matrix P. Then

$$\operatorname{sym}(\mathbf{S}_1)=\sum_k \alpha_kAB_k^{-T}\operatorname{sym}(B_k+C_k)B_k^{-1}A,$$

and each of these terms is positive definite. Now

$$\begin{aligned}
2\operatorname{sym}(\mathbf{S}_2)&=\sum_{\substack{k,j\\k\neq j}}\alpha_k\alpha_j[(AB_k^{-T}-AB_j^{-T})AB_k^{-1}A+AB_k^{-T}A(B_k^{-1}A-B_j^{-1}A)]\\
&=\sum_{\substack{k,j\\k\neq j}}\alpha_k\alpha_j[AB_k^{-T}AB_k^{-1}A-AB_k^{-T}AB_j^{-1}A-AB_j^{-T}AB_k^{-1}A+AB_j^{-T}AB_j^{-1}A]\\
&=\sum_{\substack{k,j\\k\neq j}}\alpha_k\alpha_j[(AB_k^{-T}-AB_j^{-T})A(B_k^{-1}A-B_j^{-1}A)].
\end{aligned}$$

Thus sym $(\mathbf{S}_2)$ is positive definite and the result is established.

(c) The infinity norm of a matrix is the maximum absolute row sum, and the absolute row sums of $\mathbf{H}$ are bounded by convex combinations of the absolute row sums of $B_k^{-1}C_k$. Thus $\|\mathbf{H}\|_\infty<1$, and convergence is established. □

This theorem says that if we have a collection of convergent splittings of a matrix, then under certain conditions we can construct a convergent multi-splitting. There is another way to construct convergent multi-splittings. We break the matrix into simple pieces A_k and add diagonal matrices E_k to ensure that each $B_k\equiv A_k+E_k$ is invertible.

DEFINITION. Let A_k, B_k, and D_k be $n \times n$ matrices. Then (A_k, E_k, D_k) is called a *dissolution of A* if

i) $A = \sum_k A_k$.

ii) E_k and D_k are diagonal matrices.

iii) (B_k, C_k, D_k) is a multi-splitting of A, where $B_k = A_k + E_k$ and $C_k = E_k - \sum_{j \neq k} A_j$.

We call (A_k, E_k, D_k) a *convergent dissolution of A* if it is a dissolution for which the multi-splitting (B_k, C_k, D_k) leads to a convergent algorithm.[1]

The next theorem gives explicit conditions on the matrices A, A_k, and E_k such that (A_k, E_k, D_k) will be a convergent dissolution of a matrix A.

THEOREM 2. *Let $A = \sum_k A_k$ be an M-matrix and let the matrices E_k be nonnegative diagonal matrices with diagonal components equal to e_{lk}. Then if the matrices $A_k = (a^k_{lm})$ satisfy*

(a) $0 \leqq -a^k_{lm} \leqq -a_{lm}$, $l \neq m$,

(b) $e_{lk} + a^k_{ll} > -\sum_{m \neq l} a^k_{lm}$,

(c) $e_{lk} + a^k_{ll} \geqq a_{ll}$,

then for all nonnegative diagonal matrices D_k with $\sum_k D_k = I$, (A_k, E_k, D_k) is a convergent dissolution of A.

Proof. Let us examine the elements of C_k. For $l \neq m$,

$$c^k_{lm} = -\sum_{j \neq k} a^j_{lm} = a^k_{lm} - a_{lm} \geqq 0$$

using assumption (a) and the fact that $A = \sum_k A_k$. By (c), $c^k_{ll} = e_{lk} + a^k_{ll} - a_{ll} \geqq 0$. Now $B_k = A_k + E_k$ satisfies $b^k_{lm} = a^k_{lm} \leqq 0$, $l \neq m$, by (a), and $b^k_{ll} = a^k_{ll} + e_{lk} > 0$ by (c). Further, by (b), B_k is a strictly row diagonally dominant matrix. Therefore, B_k is an M-matrix [7] and $B_k^{-1} \geqq 0$. Thus $A = B_k - C_k$ is a weak regular splitting for each k, and, by Theorem 1a, the multi-splitting is convergent. □

THEOREM 3. *Let $A = \sum_k A_k$ be a symmetric positive definite matrix, and let $A_k + E_k$ be nonsingular and $2(A_k + E_k) - A$ be positive definite, $k = 1, 2, \cdots, K$. Then for nonnegative diagonal matrices $D_k = \alpha_k I$, (A_k, E_k, D_k) is a convergent dissolution of A.*

Proof. The conditions in the theorem assure that $B_k = A_k + E_k$ is invertible and

$$B_k + C_k = A_k + 2E_k - \sum_{j \neq k} A_j = 2(A_k + E_k) - A$$

is positive definite. Convergence follows from Theorem 1b. □

3. Examples of multi-splittings. In this section we construct some examples of convergent multi-splittings of matrices. We also discuss the use of multi-splittings on parallel computers. Many other approaches to parallel iterative methods have been developed; see, for example, [2], [4], [8]. We consider the following algorithm for solving $Ax = b$. It is equivalent to Algorithm 1 when $\omega = 1$.

```
ALGORITHM 2
Choose $x_0$ arbitrarily; choose a parameter $\omega$.
For $i = 1, 2, \cdots$ until convergence
    Let $\bar{x} = x_i$.
    For $k = 1, 2, \cdots, K$
        Find $D_k y_k$ where $y_k$ satisfies
(3.1)       $B_k y_k = C_k \bar{x} + b$.
    Form $x_{i+1} = (1 - \omega) x_i + \omega \sum_k D_k y_k$.
```

[1] We use the term "dissolution" in the sense of "the breaking up of an assembly or organization" (*Random House Dictionary*, 1980).

As before, D_k is a diagonal matrix and $A = B_k - C_k$.

Parallelism in Algorithm 2 could be exploited in several ways, depending on the precise machine architecture and the choice of the multi-splitting (B_k, C_k, D_k). First, the computations in (3.1) for various k are independent, and could be performed in parallel. (Note that if a main diagonal element of D_k is zero, the corresponding component of y_k need not be computed at all.) Second, the n components of a single vector y_k (or x_{i+1}) could be computed in parallel. Third, the accumulation of the sum of $K+1$ terms which forms a component of x_{i+1} could be formed in $O(\log_2(K))$ time using parallel computation.

Our first two examples illustrate the use of convergent multi-splittings to solve algebraic systems resulting from applying the finite difference and finite element methods to partial differential equations. In both examples, the original matrix is decomposed into a sum of matrices which are considerably "simpler" than the original one and which reflect significant contributions to A from given subsets of nodes. Thus it is natural to use these decompositions of A as the basis for a dissolution as defined in § 2.

Example 1. *Decomposition by blocks of unknowns.* Consider the partial differential equation

$$-u_{xx} - u_{yy} = f \quad \text{on } \Omega, \qquad u = g \quad \text{on } \partial\Omega.$$

Let Ω be a square and use the second order accurate 5-point finite difference method to discretize the equation with m equally spaced interior mesh points in each direction. This gives the algebraic equation $Au = \bar{f}$ where A is an $m^2 \times m^2$ matrix, $\bar{f}$ is a $m^2 \times 1$ column vector whose components reflect f, g, and the dimension m, and $u = (u_{11}, \cdots, u_{1m}, \cdots, u_{m1}, \cdots, u_{mm})^T$. The matrix may be written as

$$A = \begin{bmatrix} B & -I & & & \\ -I & B & -I & & \\ & \ddots & \ddots & \ddots & \\ & & & & -I \\ & & & -I & B \end{bmatrix},$$

where

$$B = \begin{bmatrix} 4 & -1 & & & \\ -1 & 4 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & & & -1 \\ & & & -1 & 4 \end{bmatrix},$$

and is of dimension $m \times m$. This linear system can be solved, for example, by the alternating direction implicit (ADI) iterative method. One version of this method is, first, solve m sets of equations, one set for each row of mesh points, and second, solve another m sets of equations, one set for each column. That is, A is decomposed as a sum of 2 matrices:

$$A = \begin{bmatrix} T & & & \\ & T & & \\ & & \ddots & \\ & & & T \end{bmatrix} + P \begin{bmatrix} T & & & \\ & T & & \\ & & \ddots & \\ & & & T \end{bmatrix} P^T,$$

where

$$T=\begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \cdot & \vdots & \vdots & \\ & & & & -1 \\ & & & -1 & 2 \end{bmatrix}_{m\times m},$$

and P is the permutation matrix which would reorder the vector u as $Pu=(u_{11},\cdots,u_{m1},\cdots,u_{1m},\cdots,u_{mm})^T$. This can further be broken into the sum of $2m$ matrices A_k, each one corresponding to one of the matrices T. We introduce nonnegative diagonal matrices E_k such that the matrices $B_k\equiv A_k+E_k$ are invertible, and construct a set of nonnegative diagonal matrices D_k which sum to I. Since it is natural to let a diagonal component of D_k be zero if the component corresponds to a mesh point or element not in block k, most of the linear systems in Equation (3.1) of Algorithm 2 do not require the computation of a full n-dimensional problem but one whose size is much smaller—dimension m. The matrix E_k can be taken as 0 when A_k is nonzero and as arbitrary positive diagonal elsewhere. The solution of each linear system is independent of the others and can be performed in parallel if sufficient processors are available. Under natural assignments of unknowns to processors, nearby mesh points will be computed in nearby processors, so communication in step (3.1) will be local. Theorem 1a applies to this multisplitting and assures convergence.

Example 2. *Decomposition by finite elements.* Consider the Galerkin formulation of the finite element method applied to the ordinary differential equation

$$-u_{xx}=f,\qquad u(0)=u_0,\quad u(1)=u_1.$$

When linear shape functions are used on an equally spaced mesh of size $1/(m+1)$, this method gives a system of equations $Au=\bar{f}$, where A is the matrix T defined in Example 1 above. This matrix may be "assembled" by using the element matrices. The domain, $[0,1]$, is a union of $m+1$ elements $[x_i, x_i+\Delta x]$, where $m\geqq 3$. The element matrices have the form

$$\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}.$$

Then A may be written as

$$A=\sum_{k=1}^{m-1} A_k,$$

where

$$A_1=\begin{bmatrix} 2 & -1 & & & \\ -1 & 1 & & & \\ & & 0 & & \\ & & & \ddots & \\ & & & & 0 \end{bmatrix},$$

$$A_{m-1}=\begin{bmatrix} 0 & & & & \\ & \ddots & & & \\ & & 0 & & \\ & & & 1 & -1 \\ & & & -1 & 2 \end{bmatrix},$$

and A_k, $k \neq 1, m-1$ has the element matrix as a diagonal block starting in row and column k and zeros elsewhere. This splitting of finite element matrices has also been used as the basis of an iterative method by Hayes and Devloo [3]. Let $B_k = A_k + E_k$ where E_k is any diagonal matrix which makes $B_k^{-1} \geqq 0$. (For example, it is sufficient that all diagonal elements of B_k be equal to 2.) Then Theorem 1a ensures that Algorithm 2 will converge. Again, under natural assignment of nodes to processors, high parallelism can be achieved.

We have chosen trivial problems in Examples 1 and 2 to make the descriptions easier, but the methods are equally applicable to irregular meshes in several space dimensions.

We now give an example of a convergent multi-splitting which is not derived from a dissolution.

Example 3. *Decomposition by block iterative methods.* Let A be a sparse M-matrix (The matrices of the first two examples satisfy this hypothesis). Assume that each unknown has been assigned to a processor on a parallel computer. Choose some subset of K unknowns, and direct the corresponding processors to "grow" a block of unknowns from those local to it and in nearby processors in order to identify a principal submatrix of A for which linear systems are easy to solve. Note that an unknown may appear in several blocks, and the idea is to let the blocks grow to some point such that each unknown appears in at least one block and the work among processors is nearly balanced. Then, for $k = 1, 2, \cdots, K$, we have partitioned a permuted version of A as

$$\begin{bmatrix} G_{11}^k & G_{12}^k \\ G_{21}^k & G_{22}^k \end{bmatrix},$$

where G_{11}^k is the principal submatrix grown by the kth unknown. Then let

$$B_k = \begin{bmatrix} G_{11}^k & 0 \\ 0 & G_{22}^k \end{bmatrix}, \qquad C_k = \begin{bmatrix} 0 & G_{12}^k \\ G_{21}^k & 0 \end{bmatrix}, \qquad D_k = \begin{bmatrix} D_{11}^k & 0 \\ 0 & 0 \end{bmatrix},$$

where each diagonal element of $(D_{11}^k)^{-1}$ equals the number of blocks in which the corresponding unknown appears. We have a set of regular splittings of the M-matrix (because each corresponds to a block Jacobi method), and Theorem 1a assures convergence. Note that the blocks corresponding to the second row of B_k and C_k are never used since the corresponding elements of D_k are 0.

Although convergence is assured in each of these examples, it may be too slow in practice. The practical use of these algorithms in the parallel solution of sparse linear systems may be as highly parallel preconditionings of some faster iterative method such as conjugate gradients or block conjugate gradients [5].

4. Numerical examples. In the following examples we apply Algorithm 2 to two problems and study the convergence of the algorithm as the block size, the choice of E_k, and the choice of ω are changed. The second example arises from an elliptic boundary value problem and is more realistic than the first in the size and character of the resulting matrix.

Numerical example 1. Consider the ordinary differential equation

$$-u_{xx} = f(x) = 10, \qquad u(0) = 1 = u(1).$$

Let the interval [0, 1] be divided into 18 equal elements of length $h = \frac{1}{18}$, and consider the Galerkin formulation of the finite element method with linear shape functions. The resulting algebraic system is

$$\frac{1}{h}(-u_{i+1}+2u_i-u_{i-1})=\frac{h}{6}(f_{i+1}+4f_i+f_{i-1}), \qquad i=1,2,\cdots,17,$$

with $u_0=1=u_{18}$. We construct $K=16$ matrices $A_k^{(K)}$ as in Example 2 of § 3, and further consider examples in which $K=8$, $K=4$, and $K=2$. In these later examples, $A_j^{(K)}$ is formed by grouping together the elements which contributed to $A_{2j-1}^{(2K)}$ and $A_{2j}^{(2K)}$. (The case $K=1$ reduces to one iteration in which a tridiagonal problem is solved for 17 unknowns.) We choose the weighting matrices D_k to have either zero, $\frac{1}{2}$, or 1 as diagonal components:

0 if node i does not belong to finite elements in the kth set,
$\frac{1}{2}$ if node i is on the boundary of the kth set of elements,
1 if node i is in the interior of the kth set of elements.

The first choice of E_k has the form $E_k=(d/h)I$. In the last set of experiments, we used E_k defined by

$$(E_k)_{ii}=\begin{cases}0 & \text{if node } i \text{ is in the interior of block } k,\\ 1/h & \text{otherwise.}\end{cases}$$

This choice means that the diagonals of the iteration matrix $B_k=A_k+E_k$ match those of A for all components in element block k.

Tables 1-4 indicate the number of iterations required to reach convergence, defined when the relative error for each node was less than 10^{-4}. The initial guess was taken to be the vector of all one's. The values of d in Table 2 and ω in Table 3 are near optimal.

TABLE 1
Algorithm performance on Example 1 *varying K*

K	ω	d	Number of iterations
2	1.00	0.35	72
4	1.00	0.35	74
8	1.00	0.35	75
16	1.00	0.35	∞

TABLE 2
Algorithm performance on Example 1 *with* $\omega=1$ *and near optimal* $E_k=(d/h)I$

K	ω	d	Number of iterations
2	1.00	0.05	18
4	1.00	0.20	47
8	1.00	0.35	75
16	1.00	0.70	127

TABLE 3
Algorithm performance on Example 1 with $d = 1$ and near optimal $\omega = 1.32$

K	ω	d	Number of iterations
2	1.32	1.00	138
4	1.32	1.00	137
8	1.32	1.00	136
16	1.32	1.00	135

TABLE 4
Algorithm performance on Example 1 for the second choice of E_k

K	ω	Number of iterations
2	1.00	33
4	1.00	55
8	1.00	96
16	1.00	169

Numerical example 2. Consider the elliptic partial differential equation

$$-(c_1 u_x)_x - (c_2 u_y)_y = g \quad \text{on } \Omega \equiv (0,1)\times(0,1) - [\tfrac{2}{5}, \tfrac{3}{5}]\times[\tfrac{2}{5}, \tfrac{3}{5}],$$
$$u = x^2 + y^2 \quad \text{on } \partial\Omega,$$

where

$$c_1 = 1 + x^2 + y^2,$$
$$c_2 = 1 + e^x + e^y,$$
$$g = -2(2 + 3x^2 + y^2 + e^x + (1+y)e^y).$$

The data have been chosen so that the solution is $u = x^2 + y^2$. This problem is discretized by the second order accurate finite difference method with mesh spacings in both directions equal to $h = 1/(m+1)$ where $m = 9$, $m = 19$ or $m = 29$. We consider a multi-splitting as in Example 1 of § 3. Thus each A_k corresponds to some row or column of mesh points. When $m = 9$, $K = 24$ and the number of unknowns is $N = 9^2 - 3^2 = 72$. When $m = 19$, $K = 48$ and $N = 19^2 - 5^2 = 336$; when $m = 29$, $K = 72$ and

TABLE 5
Algorithm performance on Example 2

N	ω	Number of iterations
72	1.30	17
336	1.30	91
792	1.30	234

$N = 29^2 - 7^2 = 792$. Since each mesh point is involved in exactly two matrices A_k, we take the ith diagonal element of D_k equal to $1/2$ if mesh point i is in block k and 0 otherwise. Let E_k be that matrix which makes the diagonal elements of $B_k = A_k + E_k$ equal to the diagonal elements of A whose rows correspond to nodes in block k. (Numerical experiments showed that this choice led to fewer iterations than a choice of the form $E_k = (d/h^2)I$.) Convergence was defined by the l_2 norm of the discrete error vector being less than h^2. The initial guess was the zero vector. By numerical experiments, $\omega = 1.3$ was determined to be near optimal. Results appear in Table 5.

REFERENCES

[1] A. Berman and R. J. Plemmons, *Nonnegative Matrices in the Mathematical Sciences*, Academic Press, New York, 1979.

[2] V. Conrad and Y. Wallach, *Alternating methods for sets of linear equations*, Numer. Math., 32 (1979), pp. 105-108.

[3] L. J. Hayes and P. Devloo, *An overlapping block iterative scheme for finite element methods*, Dept. Aerospace Engineering and Engineering Mechanics Report, University of Texas at Austin, to appear.

[4] R. W. Hockney and C. R. Jesshope, *Parallel Computers: Architecture, Programming, and Algorithms*, Adam and Hilger, Bristol, 1981.

[5] D. P. O'Leary, *The block conjugate gradient algorithm and related methods*, Linear Algebra Appl., 29 (1980), pp. 293-322.

[6] James M. Ortega, *Numerical Analysis, A Second Course*, Academic Press, New York, 1972.

[7] A. M. Ostrowski, *Uber die Determinanten mit überwiegender Hauptdiagonale*, Comment. Math. Helv., 10 (1937), pp. 69-96.

[8] F. Robert, *Méthodes itératives série parallèle*, C.R. Acad. Sc. Paris, A-271 (1970), pp. 847-850.

[9] P. Stein, *Some general theorems on iterants*, J. Res. Nat. Bur. Standards, 48 (1952), pp. 82-83.