NASA Contractor Report 172130

ICASE

M-STEP PRECONDITIONED CONJUGATE GRADIENT METHODS

FOR REFERENCE

Loyce Adams

NOT TO BE TAKEN FROM THIS ROOM

INSTITUTE FOR COMPUTER APPLICATIONS IN SCIENCE AND ENGINEERING
NASA Langley Research Center, Hampton, Virginia 23665

Operated by the Universities Space Research Association

NASA

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665

M–STEP PRECONDITIONED CONJUGATE GRADIENT METHODS

Loyce Adams

Institute for Computer Applications in Science and Engineering

ABSTRACT

This paper describes preconditioned conjugate gradient methods for solving sparse symmetric and positive definite systems of linear equations. Necessary and sufficient conditions are given for when these preconditioners can be used and an analysis of their effectiveness is given. Efficient computer implementations of these methods are discussed and results on the CYBER 203 and the Finite Element Machine under construction at NASA Langley Research Center are included.

N83-28935 #

Introduction

In this paper we are concerned with the solution of a sparse $N \times N$ system of symmetric and positive definite linear equations

$$K\underline{u} = \underline{f} \tag{1.1}$$

by preconditioned conjugate gradient (PCG) methods. For a detailed description of these methods see Concus, Golub, O'Leary [1976] and Chandra [1978].

The PCG method solves the system, $\hat{K}\hat{\underline{u}} = \hat{\underline{f}}$, where

$$\hat{K} = Q^T M^{-1} K Q^{-T}, \quad \hat{\underline{u}} = Q^T \underline{u}, \quad \hat{\underline{f}} = Q^{-1}\underline{f}, \tag{1.2}$$

Q is a nonsingular matrix, and the symmetric and positive definite preconditioning matrix is given by $M = QQ^T$. The algorithm for the solution of $\underline{u}$ directly is described in Chandra [1978] and is given below where u, r, $\hat{r}$, and p are vectors and (x,y) denotes the inner product $x^T y$.

(1) Choose u^o

(2) $r^o = f - Ku^o$

(3) $M\hat{r}^o = r^o$

(4) $p^o = \hat{r}^o$

(5) $k = 0$

(6) For $k = 0,1,\cdots,k_{max}$

$$(1) \quad \alpha = \frac{(\hat{r}^k, r^k)}{(p^k, Kp^k)}$$

$$(2) \quad u^{k+1} = u^k + \alpha p^k$$

$$(3) \quad \text{If} \quad \|u^{k+1} - u^k\|_\infty < \varepsilon \quad \text{then stop, otherwise continue.}$$

$$(4) \quad r^{k+1} = r^k - \alpha K p^k$$

$$(5) \quad M\hat{r}^{k+1} = r^{k+1}$$

$$(6) \quad \beta = \frac{(\hat{r}^{k+1}, r^{k+1})}{(\hat{r}^k, r^k)}$$

$$(7) \quad p^{k+1} = \hat{r}^{k+1} + \beta p^k$$

Algorithm 1. Preconditioned Conjugate Gradient Algorithm.

We note that the standard conjugate gradient algorithm results by choosing $M = I$.

In the next section preconditioners that are based on taking m steps of an iterative method are described, conditions for their applicability to and effectiveness for symmetric and positive definite systems are given, and their relationship to the preconditioners of Dubois, Greenbaum, Rodrique [1979] and Johnson, Micchelli, and Paul [1982] is discussed. In Section 3, the implementation of the m-step SSOR preconditioner on parallel machines is discussed and results of this preconditioner on the CYBER 203/205 and the Finite Element Machine are included.

2. m-Step Preconditioners

2.1. Choosing M

Algorithm 1 of the last section requires a symmetric and positive definite preconditioning matrix M to be specified or computed. The question arises as how to choose M so that the condition number of $\hat{K}$,

$$\kappa(\hat{K}) = \frac{\max_i \lambda_i}{\min_i \lambda_i} ,$$

where λ_i are the eigenvalues of $M^{-1}K$, is as small as possible.

The best choice for M in the sense of minimizing $\kappa(\hat{K})$ is $M = K$ but this gains nothing since $K\hat{r} = r$ is just as difficult to solve as $Ku = f$. One approach that has been taken in the literature is to choose M to be an incomplete Cholesky factorization of K, (Manteuffel [1979]). Another approach is to choose M to be a symmetric and positive definite splitting of K that describes a linear stationary iterative method (refer to Concus, Golub, O'Leary [1976] and the references therein).

The question of interest here is whether it would be beneficial to take more than one step of a linear stationary iterative method to produce a preconditioner M that more closely approximates K. We begin by deriving an expression for M. Let $K = P - Q$ be a splitting of K that is associated with the linear stationary iterative method with iteration matrix $G = P^{-1}Q$. Then the m-step iterative method applied to $K\hat{r} = r$ is

$$P(1+G+\ldots+G^{m-1})^{-1}\hat{r}^{(m)} = \left[P(1+G+\ldots+G^{m-1})^{-1} - (P-Q)\right]\hat{r}^{(0)} + r. \qquad (2.1)$$

By choosing $\hat{r}^{(0)} = 0$, (2.1) yields

$$M = P(I+G+...+G^{m-1})^{-1}. \qquad (2.2)$$

Before we establish the necessary and sufficient conditions for M to symmetric and positive definite, we prove the following lemma.

Lemma 1.

If A = BC is a symmetric positive definite matrix, B is symmetric, and C has positive eigenvalues, then B is positive definite.

Proof

Let $C^{-1}\underline{x} = \lambda\underline{x}$ or equivalently

$$A^{-1}B\underline{x} = \lambda\underline{x} \qquad (2.3)$$

Multiply both sides by $A^{1/2}$ to get

$$\left(A^{-1/2}BA^{-1/2}\right)\left(A^{1/2}\underline{x}\right) = \lambda A^{1/2}\underline{x} \qquad (2.4)$$

or

$$R\underline{y} = \lambda\underline{y}.$$

The proof is now by contradiction. Assume that B has a non-positive eigenvalue. Then, since (2.4) is a congruency transformation of B, it follows that R has a nonpositive eigenvalue (see Gantmacher [1959]). But the spectrum of R is identical to that of C^{-1} and by hypothesis can not have a nonpositive eigenvalue. Hence B is positive definite.

The necessary and sufficient conditions for M to be positive definite are given in Theorem 1.

Theorem 1.

Let $K = P - Q$ be a symmetric positive definite matrix and let P be a symmetric nonsingular matrix. Then

(1) the matrix M of (2.2) is symmetric.

(2) for m odd, M is positive definite if and only if P is positive definite.

(3) for m even, M is positive definite if and only if $P + Q$ is positive definite.

Proof

To prove symmetry, we write M^{-1} as

$$M^{-1} = P^{-1} + P^{-1}QP^{-1} + P^{-1}QP^{-1}QP^{-1} + \cdots + \underbrace{P^{-1}QP^{-1}Q\cdots P^{-1}QP^{-1}}_{m-1 \text{ terms}} \qquad (2.5)$$

Now since P and K and hence Q are symmetric, each term in (2.5) is symmetric. Thus M is symmetric.

The matrix $G = P^{-1}Q$ can be expressed as $G = K^{-\frac{1}{2}}(I-K^{\frac{1}{2}}P^{-1}K^{\frac{1}{2}})K^{\frac{1}{2}}$. Since P^{-1} is symmetric with P, the eigenvalues of the congruence transformation $K^{\frac{1}{2}}P^{-1}K^{\frac{1}{2}}$ are real. Hence, the eigenvalues of G are real.

To prove (2), let m be odd. If g is any eigenvalue of G other than 1, the corresponding eigenvalue of

$$R = \left(I + G + \cdots + G^{m-1}\right)$$

is

$$1 + g + \cdots + g^{m-1} = \frac{1 - g^m}{1 - g}$$

which is positive since m is odd. If $g = 1$, the corresponding eigenvalue of R is equal to m and is also positive. Now, since $P = MR$ and M is symmetric and R has positive eigenvalues, it follows from Lemma 1 that if P is positive definite then M must also be positive definite. Conversely, M can be written as $M = PR^{-1}$. Since R^{-1} has positive eigenvalues and P is symmetric, we conclude from Lemma 1 that if M is positive definite then P is also positive definite.

Next, to prove (3) let m be even. It is sufficient to consider M^{-1} since any conclusions about the definiteness of M^{-1} will apply to M. Since m is even, M^{-1} from (2.5) can be written as

$$M^{-1} = P^{-1}\left(P + PG + PG^2 + PG^3 + \cdots + PG^{m-1}\right)P^{-1}$$

or

$$M^{-1} = P^{-1}\left[(P+PG) + (P+PG)G^2 + (P+PG)G^4 + \cdots + (P+PG)G^{m-2})\right]P^{-1}$$

Now, since $PG = Q$, M^{-1} can be written as

$$M^{-1} = P^{-1}(P+Q)\left(I + G^2 + G^4 + \cdots + G^{m-2}\right)P^{-1}. \tag{2.6}$$

Since P is nonsingular and symmetric, M^{-1} is positive definite if and only if the symmetric matrix

$$S = (P+Q)\left(I+G^2+G^4+\cdots+G^{m-2}\right) \qquad (2.7)$$

is positive definite.

Assume $P + Q$ is positive definite. Since S is symmetric and the matrix $\left(I+G^2+G^4+\cdots+G^{m-2}\right)^{-1}$ has positive eigenvalues, S is positive definite by Lemma 1. Conversely, if S is positive definite, since $P + Q$ is symmetric and the series $I+G^2+G^4+\cdots+G^{m-2}$ has positive eignevalues, $P + Q$ is positive definite by Lemma 1.

Dubois, Greenbaum, and Rodrique [1979] considered a truncated Neumann series for K^{-1} as a preconditioner. Their preconditioner is equivalent to that of (2.2) if $K = P - Q$ corresponds to a Jacobi splitting where $P = \text{diag}(K)$, but they do not consider more complicated splittings that result from other iterative methods. Theorem 1 extends their main result. Under the hypothesis that K and P are both symmetric and positive definite matrices and $p(G) < 1$, they prove that M is symmetric and positive definite for all m. Note that for odd m the condition that $p(G) < 1$ is not needed. The relationship between the condition $p(G) < 1$ and the positive definiteness of $P + Q$ is given in Theorem 2.

Theorem 2.

Let $K = P - Q$ be a symmetric positive definite matrix and let P be symmetric and nonsingular. Then $\rho\left(P^{-1}Q\right) < 1$ if and only if $P + Q$ is positive definite.

<u>Proof</u>

First, assume $P + Q$ is positive definite. Since K is symmetric positive definite and P is nonsingular, $K = P - Q$ is a p-regular splitting. Hence, from Ortega's p-regular splitting theorem, Ortega [1972], $\rho(P^{-1}Q) < 1$.

Now, assume that $\rho(G) < 1$. Then $(I-G)^{-1}$ exists and since G has real eigenvalues, it easily follows that the matrix H defined by

$$H = (I-G)^{-1}(I+G) \tag{2.8}$$

has real eigenvalues. But we know from Young ([1971], p. 82) that H is N-stable. Hence H has positive eigenvalues. Now, we can write H as

$$H = K^{-1}(P+Q) \tag{2.9}$$

or equivalently,

$$K = (P+Q)H^{-1} \tag{2.10}$$

Finally, since K is symmetric and positive definite and H^{-1} has positive eigenvalues and $P + Q$ is symmetric, we conclude from Lemma 1 that $P + Q$ is positive definite.

We note that the Jacobi Convergence Theorem given in Young [1971] is a specific case of Theorem 2.

Theorem 1 and Theorem 2 are helpful in choosing a splitting of K that will produce an m-step preconditioner that is symmetric and positive

definite. For example, if the Jacobi splitting of K (P = D and Q = D - K where D is the diagonal of K) were considered, part (3) of Theorem 1 says that if m is even, P + Q must be positive definite, and by Theorem 2 this is only true when the Jacobi method is convergent. However, for problems of interest to us, the Jacobi method is not guaranteed to be convergent since we only know that K will be symmetric and positive definite; therefore, for these problems, only odd values of m will yield m-step Jacobi preconditioning matrices that are guaranteed to be positive definite.

2.2. Analysis of the Condition Number

In the last section, we gave conditions for M to be symmetric and positive definite and hence to be considered as a preconditioner for the conjugate gradient method. In this section we determine if increasing m will, in fact, produce a better conditioned system. For this purpose, we now denote by M_m the matrix of (2.2).

As a first step towards answering this question, we derive an expression for $\kappa(\hat{K}_m)$. Recall from (1.2) that $\hat{K}$ is similar to $M_m^{-1}K$ so that $\kappa(\hat{K}_m)$ is the same as the ratio of the largest to smallest eigenvalue of $M_m^{-1}K$. An expression for $M_m^{-1}K$ as a polynominal in G is

$$M_m^{-1}K = \left(I+G+\ldots G^{m-1}\right)P^{-1}(P-Q) \qquad (2.11)$$

or

$$M_m^{-1}K = I - G^m$$

where $G = P^{-1}Q$.

We wish to compare $\kappa(\hat{K}_m)$ to $\kappa(\hat{K}_{m+1})$, when both M_m and M_{m+1} are symmetric and positive definite. By Theorem 1, this implies that P and P + Q are positive definite and thus by Theorem 2, $\rho(G) < 1$. Under the

hypothesis of Theorem 1 the eigenvalues λ_i of G are real, and can be ordered as

$$-1 < \lambda_1 < \lambda_2 < \cdots < \lambda_n < 1.$$

Furthermore, let δ be the eigenvalue with the smallest absolute value. Then the condition number of $\hat{K}_m$ is

$$\kappa\left(\hat{K}_m\right) = \begin{cases} \dfrac{1 - \lambda_1^m}{1 - \lambda_n^m} & \lambda_1 > 0 \text{ or } \lambda_1 < 0 \text{ and } m \text{ odd} \\[3mm] \dfrac{1 - \delta^m}{1 - \lambda_n^m} & \lambda_1 < 0, \ \lambda_n > |\lambda_1|, \ m \text{ even} \\[3mm] \dfrac{1 - \delta^m}{1 - \lambda_1^m} & \lambda_1 < 0, \ |\lambda_1| > |\lambda_n|, \ m \text{ even} \end{cases} \qquad (2.12)$$

As can be seen from (2.12), the conditions for $\kappa\left(\hat{K}_{m+1}\right) < \kappa\left(\hat{K}_m\right)$ depend upon the distribution of the eigenvalues λ_i of G. We note that for both odd and even m if $\lambda_1 < 0$ and $|\lambda_1| > |\lambda_n|$, it is impossible to decide whether $\kappa\left(\hat{K}_{m+1}\right) < \kappa\left(\hat{K}_m\right)$ without knowledge of the values of λ_1, λ_n, and δ. The conditions for the remaining two cases are stated below:

If $\lambda_1 > 0$, $\kappa\left(\hat{K}_m\right)$ is a decreasing function for all m. $\qquad (2.13a)$

If $\lambda_n > |\lambda_1|$ and $\lambda_1 < 0$,
    (a) for m odd, $\kappa\left(\hat{K}_{m+1}\right) < \kappa\left(\hat{K}_m\right)$.

$\qquad (2.13b)$

    (b) for m even, $\kappa\left(\hat{K}_{m+1}\right) < \kappa\left(\hat{K}_m\right)$ if and only if

$$\left(1 + |\lambda_1|^{m+1}\right)\left(1 - \lambda_n^m\right) < \left(1 - \delta^m\right)\left(1 - \lambda_n^{m+1}\right).$$

As an application of (2.13a) consider the SSOR splitting of a symmetric and positive definite matrix. Recall from the basic convergence theorem for SSOR that if K is a symmetric matrix with positive diagonal elements, the SSOR method converges if and only if K is positive definite and $0 < \omega < 2$. Therefore, $\rho(G) < 1$ for this splitting and from Young [1971] we know that all the eigenvalues of G are real and nonnegative. Since P is symmetric, it follows from Theorems 1 and 2 that M_m is symmetric and positive definite and from (2.13a) it follows that $\kappa(\hat{K}_m)$ is a decreasing function of m.

Results of the m-step SSOR preconditioned conjugate gradient method on a 1536×1536 symmetric and positive definite matrix derived from a finite element discretization (triangles with linear basis functions) of a plate in plane stress are given in Table I and the results on a 768×768 matrix derived from the 5-star discretization of Laplace's equation are given in Table II. For these problems, results are given for both the natural rowwise ordering and the Multi-color ordering (see Adams and Ortega [1982]) of the grid. The convergence criterion was $\|\underline{u}^{k+1} - \underline{u}^k\|_\infty < \varepsilon$, where $\varepsilon = 10^{-6}$ for both problems. The conjugate gradient results with no preconditioning are indicated by $m = 0$.

Table I. m-step SSOR PCG for 1536 × 1536 Plane Stress Problem

m	R/B/G # Iterations ($\omega=1$)	Natural # Iterations ($\omega=1$)	# Iterations ($\omega=1.6$)
0	363	363	363
1	139	111	93
2	99	80	66
3	82	65	54
4	71	57	47

Table II. m-step SSOR PCG for 768 × 768 Laplace's Equation

m	R/B # Iterations ($\omega=1$)	Natural # Iterations ($\omega=1$)	# Iterations ($\omega=1.8$)
0	56	56	56
1	30	28	17
2	22	21	13
3	18	17	10
4	16	15	9

The results in Tables I and II show that the number of iterations is a decreasing function of m as was predicted by (2.13a). The results also indicate that there will be an optimal value of m, say m_{opt}; since for m > m_{opt}, the reduction in the number of CG iterations is not enough to balance the increase in the time required for the iterations of the SSOR preconditioner. The actual relative cost of the CG and SSOR iterations on a computer will be a function of the amount of arithmetic and communication

operations in each algorithm as well as the times to perform these operations on the machine. Therefore, the optimal value of m will depend on the architecture of the machine and the problem size as indicated by the results in Section 3.

As an example of an application of (2.13b) we consider the Jacobi splitting of any symmetric and positive definite matrix K that has Property A (see Young [1971]). For this splitting, P = D where D is the diagonal of K and therefore P is symmetric and positive definite. Now, since K has Property A, the eigenvalues λ_i of G occur in $\pm\lambda_i$ pairs and $\lambda_n = |\lambda_1|$ and $\delta = 0$. From (2.13b) we conclude that going from m (even) to m + 1 (odd) is advantageous if and only if

$$\left(1+\lambda_n^{m+1}\right)\left(1-\lambda_n^m\right) < \left(1-\lambda_n^{m+1}\right)$$

or equivalently, (2.14)

$$\lambda_n^{m+1} - 2\lambda_n + 1 > 0.$$

As m increases the inequality in (2.14) reduces asymptotically to

$$\lambda_n < \frac{1}{2}.$$ (2.15)

For m = 2 and m = 3, the exact conditions are $\lambda_n < .62$ and $\lambda_n < .53$ respectively, but for problems of interest to us, λ_n will be closer to 1 and we can conclude that it is not advantageous to increase m from m (even) to m + 1 (odd). This fact has been verified by numerical experiments for the m-step Jacobi preconditioner on an 89 × 89 symmetric and positive definite system that had Property A. The results are given in Table III.

Table III. m-step Jacobi Results 89 × 89

m	# Iterations
0	45
1	45
2	23
3	36
4	21
5	30
6	18
7	26
8	16

Note from Table III that increasing m from 2 to 3, from 4 to 5, and from 6 to 7 also increases the number of iterations from 23 to 36, from 21 to 30, and from 18 to 26 respectively. On the other hand, observe that increasing m from an odd to a consecutive even number always reduces the number of iterations. Dubois, Greenbaum, Rodrique [1979] reported similar results for Poisson's equation. Their results may also be explained by (2.13b). Also note from Table III that the number of iterations is a decreasing function of m if we restrict m to be even. In fact this can easily be shown to be true for all three cases in (2.12).

So far we have only addressed the question of whether a better conditioned system results by increasing m. We now turn to the question of how much improvement over m = 1 can be made by taking m > 1 steps of the preconditioner. Dubois, Greenbaum, and Rodrique [1979] proved that the m-step PCG method can only reduce the number of iterations needed by the 1-step PCG method by a factor of m. In practice, this theoretical bound may

not be reached and for a given distribution of eigenvalues it may be sharper for some values of m than for others. The results of Dubois, et.al. [1979] show this for the m-step Jacobi PCG for Laplace's equation. Tables I and II show for the m-step SSOR PCG method applied to both the plane stress problem and Laplace's equation that the bound is best for m = 2. Table III shows that for the m-step Jacobi PCG applied to a problem with Property A that the bound is extremely sharp for m = 2 and extremely poor for odd values of m.

In order to determine the conditions under which the m-step PCG method gives the most improvement over the 1-step PCG method, we examine the ratio $\frac{\kappa(\hat{K}_1)}{\kappa(\hat{K}_m)}$ for both odd and even m with different assumptions about the distribution of the eigenvalues λ_i of G which are assumed to be ordered as $-1 < \lambda_1 < \lambda_2 < ... < \lambda_n < 1$ with $\delta = \min_i |\lambda_i|$. This ratio can easily be calculated from the equations of (2.12) and is summarized below for the various cases.

16

$$\frac{\kappa(\hat{K}_1)}{\kappa(\hat{K}_m)} = \begin{cases} \dfrac{1 + \lambda_n + \lambda_n^2 + \ldots + \lambda_n^{m-1}}{1 + \lambda_1 + \lambda_1^2 + \ldots + \lambda_1^{m-1}} & \lambda_1 > 0 \\[2em] \dfrac{\left(1+|\lambda_1|\right)\left(1+\lambda_n+\lambda_n^2+\ldots+\lambda_n^{m-1}\right)}{1 + |\lambda_1|^m} & \lambda_1 < 0, \lambda_n > 0, \quad m \text{ odd} \\[2em] \dfrac{\left(1+|\lambda_n|^m\right)\left(1+|\lambda_1|\right)}{\left(1+|\lambda_1|^m\right)\left(1+|\lambda_n|\right)} & \lambda_1 < 0, \lambda_n < 0, \quad m \text{ odd} \\[2em] \dfrac{\left(1+|\lambda_1|\right)\left(1+\lambda_n+\lambda_n^2+\ldots+\lambda_n^{m-1}\right)}{\left(1-|\delta|^m\right)} & \lambda_1 < 0, \lambda_n > |\lambda_1| \quad m \text{ even} \\[2em] \dfrac{\left(1+|\lambda_1|\right)\left(1-|\lambda_1|^m\right)}{\left(1-\lambda_n\right)\left(1-|\delta|^m\right)} & \lambda_1 < 0, |\lambda_1| > |\lambda_n| \quad m \text{ even.} \end{cases} \quad (2.16)$$

Several observations can be made from (2.16):

(1) If $\lambda_1 > 0$, the maximum value of $\dfrac{\kappa(\hat{K}_1)}{\kappa(\hat{K}_m)}$ occurs as $\lambda_1 \to 0$ and $\lambda_n \to 1$ and is equal to m. (This is the case for the SSOR splitting.)

(2) If $\lambda_1 < 0$ and $\lambda_n > 0$, and m is odd, the maximum value of $\dfrac{\kappa(\hat{K}_1)}{\kappa(\hat{K}_m)}$ occurs when $\lambda_n \to 1$ and is equal to $m\left(\dfrac{1+|\lambda_1|}{1+|\lambda_1|^m}\right)$.

(3) The m-step PCG method $(m>1)$ is more effective if $\lambda_n > 0$.

(4) If $\lambda_1 < 0$, and $\lambda_n > |\lambda_1|$, and m is even, the maximum value of

$\dfrac{\kappa(\hat{K}_1)}{\kappa(\hat{K}_m)}$ occurs when $\lambda_n \to 1$ and $|\lambda_1| = |\lambda_n|$ and is equal to $\dfrac{2m}{1 - \delta^m}$.

Note that the larger δ, the larger this ratio will be. Hence to achieve the maximum performance in this case, we would like the value of δ to be as close to λ_1 as possible. For K matrices with Property A, this is not possible since $\delta = 0$ and the maximum ratio of the two condition numbers is $2m$.

In summary, the m-step PCG method gives more improvement over the 1-step PCG method when an even number of steps of the preconditioner are taken and the eigenvalues of the matrix G are distributed as described in (4) above. This implies that for the SSOR iteration matrix which has $\lambda_1 > 0$, the m-step SSOR preconditioner will not be extremely effective as m increases. However, by parametrizing this preconditioner the method is more effective. This is the topic of the next section.

2.3 Parametrizing the m-step PCG Method

Johnson, Micchelli, and Paul [1982] have suggested symmetrically scaling the matrix K to have unit diagonal and then taking m terms of a parametrized Neumann series for $K^{-1} = (I-G)^{-1}$ as the value for M^{-1}. This corresponds to a symmetric preconditioning matrix that is a polynominal of degree m-1 in G,

$$M_m^{-1} = \alpha_0 I + \alpha_1 G + \alpha_2 G^2 + \ldots + \alpha_{m-1} G^{m-1} \qquad (2.17)$$

derived from the Jacobi splitting,

$$K = I - G \qquad (2.18)$$

of K; hence, the solution to $M_m \hat{\underline{r}} = \underline{r}$ can be implemented by taking m steps of the Jacobi iterative method applied to $K\hat{\underline{r}} = \underline{r}$ with initial guess $\hat{\underline{r}}^{(o)} = \underline{0}$.

Now, $M_m^{-1}K$ can be written as a polynominal in K,

$$M_m^{-1}K = \left[\alpha_0 I + \alpha_1(I-K) + \alpha_2(I-K)^2 + \ldots + \alpha_{m-1}(I-K)^{m-1}\right]K \qquad (2.19)$$

and Johnson, et.al., choose the α_1's so that the eigenvalues of $M_m^{-1}K$, and hence those of M_m, are positive on the interval $\left[\lambda_1, \lambda_n\right]$ that contains the eigenvalues of K and are as close to 1 as possible in some sense such as the min-max or the least squares criteria. Clearly, if $m = 1$, $M_m^{-1}K = \alpha_0 K$ and the condition number of $M_m^{-1}K$ is the same for all $\alpha_0 \neq 0$. Hence, we are only interested in $m > 1$.

We now generalize this idea for any splitting of the matrix K,

$$K = P - Q. \qquad (2.20)$$

If $G = P^{-1}Q$, then by parametrizing (2.2), the inverse of the m-step preconditioner becomes

$$M_m^{-1} = \left(\alpha_0 I + \alpha_1 G + \alpha_2 G^2 + \ldots + \alpha_{m-1} G^{m-1}\right)P^{-1} \qquad (2.21)$$

and will be symmetric if P is symmetric. The expression for $M_m^{-1}K$ is given by

$$M_m^{-1}K = \left[\alpha_0 I + \alpha_1\left(I - P^{-1}K\right) + \ldots + \alpha_{m-1}\left(I - P^{-1}K\right)^{m-1}\right]P^{-1}K \qquad (2.22)$$

and is seen to be a polynominal in $P^{-1}K$ rather than in K as in (2.19). We now choose the values of α_i so that the eigenvalues of $M_m^{-1}K$ are positive on the interval $[\lambda_1, \lambda_n]$ that contains the eigenvalues of $P^{-1}K$ and are as close to 1 as possible in some sense such as the min-max or least squares criteria.

When the eigenvalues of G are on the interval $[0,1)$, the eigenvalues of $P^{-1}K$ are on the interval $(0, 1]$ and from (2.22), in the least squares sense, we wish to find the α_i's that minimize

$$\int_0^1 [\alpha_0 x + \alpha_1 (1-x)x + \ldots + \alpha_{m-1}(1-x)^{m-1}x - 1]^2 dx.$$

The appropriate values of the $\alpha_i, i = 0,1,\cdots,m-1$ for the SSOR splitting are given in Adams [1983]. In the next section we discuss the efficient implementation of the m-step SSOR preconditioner and the choice for the relaxation parameter ω for the SSOR method if the grid points are ordered by a Multi-color ordering.

3. Implementation and Results

3.1. Implementation Considerations

In order to efficiently implement the m-step SSOR preconditioner on parallel computers, the equations at the grid points of the problem domain must be colored, see Adams and Ortega [1982], so that any two equations at points on the same grid point stencil are different colors. The equations are then ordered by colors with the equations of the same color being ordered left to right, top to bottom (for a rectangular grid). In particular, if three colors are used, the system $K\hat{\underline{r}} = \underline{r}$ has the decoupled form,

$$\begin{bmatrix} D_{11} & B_{12} & B_{13} \\ B_{12}^T & D_{22} & B_{23} \\ B_{13}^T & B_{23}^T & D_{33} \end{bmatrix} \begin{bmatrix} \hat{r}_1 \\ \hat{r}_2 \\ \hat{r}_3 \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} \qquad (3.1)$$

where D_{ii}, $i = 1$ to 3 are diagonal matrices.

The m-step SSOR iteration is implemented as a forward followed by a backward Multi-color SOR iteration (Adams and Ortega [1982]) but care is taken to save results from the forward pass in an auxilary vector to be used in the reverse pass so that the cost of one SSOR iteration is no more expensive than the cost of one SOR iteration (Conrad and Wallach [1979]). Specific details on this implementation (in conjunction with Algorithm 1) for the CYBER 203 and the Finite Element Machine can be found in Adams [1983].

In addition to the computational work saved by using the auxilary vector, the Multi-color ordering permits even more savings. To explain this, we begin by writing a 3-color SOR iteration matrix, $\mathcal{L}_\omega$, in the following factored form:

$$\mathcal{L}_\omega = G_\omega B_\omega R_\omega \qquad (3.2)$$

where R_ω, B_ω, and G_ω are the matrix operators for the Red, Black, and Green equations respectively. Nicolaides [1974] discussed the factorization of an $n \times n$ SOR iteration matrix $\mathcal{L}_\omega$ into n operator matrices, one for each equation, and then showed how these factors combine for matrices with Property A into two factors, $\mathcal{L}_\omega = B_\omega R_\omega$, corresponding to the red and black equations respectively. Young [1971] also gives the factorization of $\mathcal{L}_\omega$ for these 2-colored matrices. Equation (3.2) is a straightforward continuation of these ideas. To be precise, if the matrix K is given by

$$K = \begin{bmatrix} I_1 & -X_{12} & -X_{13} \\ -X_{12}^T & I_2 & -X_{23} \\ -X_{13}^T & -X_{23}^T & I_3 \end{bmatrix} , \qquad (3.3)$$

with no loss in generality by assuming $D = I$ on the diagonal, the R_ω, B_ω, and G_ω matrices in (3.2) are

$$R_\omega = \begin{bmatrix} (1-\omega)I_1 & \omega X_{12} & \omega X_{13} \\ 0 & I_1 & 0 \\ 0 & 0 & I_2 \end{bmatrix} , \qquad B_\omega = \begin{bmatrix} I_1 & 0 & 0 \\ \omega X_{12}^T & (1-\omega)I_2 & \omega X_{23} \\ 0 & 0 & I_3 \end{bmatrix}$$

and

$$G_\omega = \begin{bmatrix} I_1 & 0 & 0 \\ 0 & I_2 & 0 \\ \omega X_{12}^T & \omega X_{23}^T & (1-\omega)I_3 \end{bmatrix} , \qquad (3.4)$$

respectively.

Similarly, the backward Multi-color SOR iteration matrix may be written in the factored form

$$\mathcal{U}_\omega = R_\omega B_\omega G_\omega \qquad (3.5)$$

where $R_\omega, B_\omega, G_\omega$ are the same as those of (3.2). Now, the Multi-color SSOR iteration matrix may be written as,

$$\mathcal{S}_\omega = R_\omega B_\omega G_\omega G_\omega B_\omega R_\omega . \qquad (3.6)$$

A trivial calculation shows that $G_\omega G_\omega = G_{\omega(2-\omega)}$ and $R_\omega R_\omega = R_{\omega(2-\omega)}$. Hence,

$$\mathscr{S}_\omega = R_\omega B_\omega G_{\omega(2-\omega)} B_\omega R_\omega. \tag{3.7}$$

From (3.7), we see that the green equations only need to be calculated on the forward pass with relaxation factor $\omega' = \omega(2-\omega)$. Likewise, the R_ω operators combine from the backward pass and the next forward pass so that the red equations should be updated on the first forward pass with relaxation factor ω and on the last backward pass with relaxation factor ω. For the intermediate forward passes, the red equations should be updated with $\omega' = \omega(2-\omega)$. The black equations, however, must be updated on both the forward and backward passes with relaxation parameter ω but part of this calculation can be saved by the use of the auxilary vector mentioned earlier. By organizing the computation in this fashion, $2m(c-1)+1$ rather than $2mc$ operation matrices need to be applied. Also, this computational organization is not affected by the introduction of $\alpha_i, i=1,2,\cdots,m$ since the parameter α_i multiplies only the right hand side vector $\underline{r}$ on step $m-i+1$ of the preconditioner.

We now briefly discuss the choice for ω. From Young's [1971] theory of matrices with Property A (2-colored) we know that the optimal ω for SSOR is $\omega = 1$. In fact, Young's proof shows that

$$\mathscr{S}_\omega = R_\omega B_\omega B_\omega R_\omega \tag{3.8}$$

and

$$\mathscr{S}_\omega \sim B_{\omega(2-\omega)} R_{\omega(2-\omega)} = \mathscr{L}_{\omega(2-\omega)} \tag{3.9}$$

and for matrices with Property A, $\mathscr{L}_{\omega(2-\omega)}$ has the smallest spectral radius whenever $\omega = 1$. In particular, $\mathscr{S}_1 \sim \mathscr{L}_1$. Now, for Multi-color matrices, $\mathscr{S}_\omega$ is not necessarily similar to $\mathscr{L}_{\omega(2-\omega)}$ since from (3.7) with 3

colors, we see that

$$\mathscr{S}_\omega \sim {}^B\omega {}^G{}_{\omega(2-\omega)} {}^B\omega {}^R{}_{\omega(2-\omega)} \tag{3.10}$$

and for $\omega = 1$,

$$\mathscr{S}_1 \sim {}^B{}_1 \mathscr{L}_1 \tag{3.11}$$

In general when the number of colors is equal to c and $C_\omega^{(k)}$ denotes the matrix associated with color k,

$$\mathscr{S}_\omega \sim C_\omega^{(2)} C_\omega^{(3)} \cdots C_\omega^{(c-1)} C_{\omega(2-\omega)}^{(c)} C_\omega^{(c-1)} \cdots C_\omega^{(3)} C_\omega^{(2)} C_{\omega(2-\omega)}^{(1)} \tag{3.12}$$

and for $\omega = 1$,

$$\mathscr{S}_1 \sim C_1^{(2)} C_1^{(3)} \cdots C_1^{(c-1)} \mathscr{L}_1 . \tag{3.13}$$

Assume that (3.12) represents an equal number of equations of each color and let $\omega > 1$ so that $\omega(2-\omega) < 1$. For two colors, (3.9) shows that all equations are underrelaxed. For three colors, (3.10) shows that we can regard only the black equations as being overrelaxed (once on the forward and once on the reverse pass). In general, (3.12) shows that the equations of $c-2$ colors can be regarded as overrelaxed and the equations of 2 colors as underrelaxed. When the number of colors approaches the number of equations, all but two equations can be regarded as being overrelaxed. Although not a proof, this observation suggests that overrelaxation becomes more worthwhile as the number of colors increases and choosing $\omega = 1$ when a small number of colors is used is a good choice. This was the case for the results in Table I, where for the R/B/G ordering of nodes (really six colors -- two unknowns per node) $\omega = 1$ was optimal for m-step SSOR PCG. Results in Adams [1982] show that $\omega = 1$ was also optimal for the SSOR method (used alone) for this same problem.

3.2. Results on Parallel Computers

We now give results of the m-step SSOR PCG method for a square plate in plane stress on both the CYBER 203 and the Finite Element Machine. These results were discussed in detail in Adams [1983] and are only included here to show that the method is effective on these machines. Table IV gives the number of iterations, I, and the time, T, in seconds to solve this problem using m = 0,1,2,3,4,5,6,7,8,9, and 10. The parametrized preconditioner results are denoted by P, the number of rows in the plate by a, and the maximum vector length by v.

Table IV.   CYBER 203 Iterations and Timings m-step SSOR PCG

m	v = 22, a = 8		v = 41, a = 11		v = 132, a = 20		v = 561, a = 41		v = 1282, a = 62		v = 2134, a = 80	
	I	T	I	T	I	T	I	T	I	T	I	T
0	112	.133	157	.213	271	.565	536	3.293	788	11.845	929	22.780
1	52	.129	66	.184	111	.454	214	2.373	311	7.832	395	17.194
2	38	.143	50	.208	79	.478	152	2.428	221	7.773	280	17.380
2P	31	.116	40	.167	61	.369	118	1.885	172	6.052	218	13.534
3	31	.155	39	.216	65	.520	124	2.585	181	8.174	229	18.469
3P	24	.121	30	.167	46	.369	88	1.836	129	5.828	163	13.151
4P	22	.138	24	.166	35	.350	67	1.726	99	5.471	124	12.306
5P	19	.143	20	.167	29	.347	56	1.716	82	5.345	104	12.260
6P	18	.159	18	.175	25	.348	47	1.670	70	5.263	88	12.011
7P					26	.413	43	1.739	64	5.451	80	12.410
8P					21	.375	36	1.634	54	5.139	69	11.985
9P							33	1.660	48	5.056	61	11.731
10P							31	1.709	44	5.070	55	11.594

We now give the Finite Element Machine results. The same problem with 6 rows and 6 columns of nodes (60 equations) was solved on a 1, 2, and then on a 5-processor Finite Element Machine using the m-step SSOR PCG method (as more processors become available on this machine the solution of larger problems will be possible). Each processor was assigned equations at an equal number of R, B, and G nodes. Therefore, in the absence of communication time and any differences in processor speeds, a speedup of 2 (5) over the one processor case should be realized whenever 2 (5) processors are used respectively. The number of iterations, I, and the time, T, in seconds as well as the respective speedups are given in Table V.

Table V. FEM Iterations, Timings, Speedups m-step SSOR PCG

	p = 1		p = 2			p = 5		
m	I	T	I	T	Speedup	I	T	Speedup
0	48	63.35	49	33.70	1.92	48	17.70	3.58
1	19	47.90	19	25.85	1.85	19	14.85	3.23
2	13	48.75	13	26.65	1.83	13	15.50	3.15
2P	11	41.95	11	22.95	1.83	11	13.30	3.15
3	11	54.95	11	30.15	1.82	11	17.65	3.11
3P	8	41.25	8	22.75	1.81	8	13.25	3.11
4	10	62.40	10	34.30	1.82	10	20.20	3.09
4P	6	39.80	6	22.00	1.81	6	12.90	3.09
5P	5	40.60	5	22.50	1.80	5	13.25	3.06
6P	5	47.05	5	26.20	1.80			

4. Summary and Conclusions

Preconditioners for a symmetric and positive definite system of linear equations based on taking m steps of an iterative method that is derived from a symmetric splitting of the coefficient matrix have been described. Necessary and sufficient conditions were given for these preconditioners to be symmetric and positive definite for both m odd and even in Theorem 1, and the relationship between a splitting and its associated iteration matrix was given in Theorem 2.

The m-step SSOR preconditioner was shown to lead to a system whose condition number was a decreasing function of m; however, for small problems, the actual decrease in the number of iterations is not enough to balance the extra work involved in the preconditioner as shown in Tables IV and V. By parametrizing this preconditioner, the number of iterations is reduced enough so that larger values of m should be used for smaller problems as well. The optimal number of steps of the preconditioner is seen from Tables IV and V to be a function of the architecture as well as the problem. The more expensive the inner products of the outer CG iteration become, the more likely m should be increased.

We noted that although a theoretical optimal value of ω, the relaxation parameter for the SSOR method, can not be found, the choice $\omega = 1$ (when the nodes are ordered by the Multi-color ordering) was optimal for our plane stress test problem (6 colors). It is well known that $\omega = 1$ is optimal for SSOR for matrices that have Young's Property A (Red/Black), but in general this theory does not extend beyond two colors. However, we conjectured that if the number of colors is small, choosing $\omega = 1$ is a good choice.

A problem still remains in applying the method to irregular regions since the grid must be colored and for array machines must also be distributed to the processors in light of this coloring.

ACKNOWLEDGEMENT

The author would like to thank Professor James M. Ortega for his useful discussions and suggestions concerning this manuscript.

REFERENCES

[1] L. ADAMS and J. ORTEGA, A multi-color SOR method for parallel computation, Proc. 1982 Conference on Parallel Processing, Bellaire, MI, August 1982, pp. 53-58.

[2] L. ADAMS, Iterative algorithms for large sparse linear systems on parallel computers, Ph.D. Thesis, University of Virginia, Department of Applied Mathematics and Computer Science, January 1983. Also NASA Contractor Report 166027, NASA Langley Research Center, 1982.

[3] L. ADAMS, An m-step preconditioned conjugate gradient method for parallel computation, to appear in 1983 IEEE Parallel Computation Conference Proceedings, Bellaire, MI, August 1983.

[4] R. CHANDRA, Conjugate gradient methods for partial differential equations, Ph.D. Thesis, Research Report #129, Department of Computer Science, Yale University, 1978.

[5] P. CONCUS, G. GOLUB, and D. O'LEARY, A generalized conjugate gradient method for the numerical solution of elliptic partial differential equations, STAN-CS-76-533, Computer Science Department, Stanford University, 1976.

[6] V. CONRAD and Y. WALLACH, Alternating methods for sets of linear equations, Numer. Math., Vol. 32 (1979), pp. 105-108.

[7]   P. DUBOIS, A. GREENBAUM, and G. RODRIQUE, Approximating the inverse of a matrix for use in iterative algorithms on vector processors, Computing, Vol. 22 (1979), pp. 257-268.

[8]   F. GANTMACHER, The Theory of Matrices, Vol. 1, Chelsea Publishing Company, New York, 1959.

[9]   O. JOHNSON, C. MICCHELLI, and G. PAUL, Polynominal preconditioners for conjugate gradient calculations, IBM Research Report #40444#, IBM Thomas J. Watson Research Center, Yorktown Heights, N.Y., 1982.

[10] T. MANTEUFFEL, An incomplete factorization technique for positive definite linear systems, Math. Comp., Vol. 34, No. 150 (1979), pp. 473-479.

[11] R. NICOLAIDES, On a geometrical aspect of SOR and the theory of consistent ordering for positive definite matrices, Numer. Math. (1974), pp. 99-104.

[12] J. ORTEGA, Numerical Analysis: A Second Course, Academic Press, New York, 1972.

[13] D. YOUNG, Iterative Solution of Large Linear Systems, Academic Press, New York, 1971.

1. Report No. NASA CR-172130	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle M-STEP PRECONDITIONED CONJUGATE GRADIENT METHODS		5. Report Date April 1983
		6. Performing Organization Code
7. Author(s) Loyce Adams		8. Performing Organization Report No. 83-9
		10. Work Unit No.
9. Performing Organization Name and Address Institute for Computer Applications in Science & Engineering Mail Stop 132C, NASA Langley Research Center Hampton, VA 23665		11. Contract or Grant No. NAG1-46,NAS1-15810,NAS1-17070 NAS1-17130
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546		13. Type of Report and Period Covered contractor report
		14. Sponsoring Agency Code

15. Supplementary Notes

Langley Technical Monitor: Robert H. Tolson

16. Abstract

This paper describes preconditioned conjugate gradient methods for solving sparse symmetric and positive definite systems of linear equations. Necessary and sufficient conditions are given for when these preconditioners can be used and an analysis of their effectiveness is given. Efficient computer implementations of these methods are discussed and results on the CYBER 203 and the Finite Element Machine under construction at NASA Langley Research Center are included.

17. Key Words (Suggested by Author(s)) preconditioned conjugate gradient, SSOR (symmetric successive overrelaxation), multi-color ordering	18. Distribution Statement Unclassified-Unlimited Subject Category 64

19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 31	22. Price A03

End of Document