J. Nig Ass. Math. Phys.
Vol. 3 (1999)

MULTIPOINT ITERATION METHODS FOR SPARSE SYSTEM OF LINEAR EQUATIONS.

IKHILE, M.N.O.
DEPARTMENT OF MATHEMATICS,
UNIVERSITY OF BENIN, P.M.B. 1154, UGBOWO CAMPUS
BENIN CITY, NIGERIA.

ABSTRACT

In this report we investigate the effects of incorporating past information in the Gauss-Jacobi iteration method for large sparse linear system of equations. It is considered that under certain restrictions convergence is assured, but the implementation requires extra-storage locations. This may not be a serious handicap considering the enormous storage capacity of emerging parallel computers. More consoling is the fact that the resultant algorithms possesses explicit potential for parallelism, although the exploitation of these is not our purpose herein. However, our findings raises the question as to whether there is any need pursuing multipoint iteration methods for sparse linear system of equations as a separate method.

1. INTRODUCTION

In real life applications especially in the solution of partial differential equations, it is often the case to solve on a large scale the linear system of equations [1,2,3]

$$Ax = b, \qquad \det(A) \neq 0 \tag{1.1}$$

where

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & & \\ a_{1n} & a_{n2} & \cdots & a_{nn} \end{pmatrix}, \quad x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \tag{1.2}$$

with the coefficient matrix **A** being sparse and with some other desirable qualities. One of the classes of successful methods employed for the solution of (1.1) is mainly iterative and the Gauss-Jacobi method, [4,5,6,7]

$$Dx^{(K+1)} = b - Mx^{(K)}; \; K = 0, 1, 2, \ldots, \tag{1.3}$$

is typical of such methods, where

$$M = \begin{pmatrix} O & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & O & a_{23} & \cdots & a_{2n} \\ \vdots & & & & \vdots \\ a_{n1} & \cdots & & a_{nn-1} & 0 \end{pmatrix}, \quad D = \begin{pmatrix} a_{11} & O & O & \cdots & O \\ O & a_{22} & O & \cdots & O \\ \vdots & & & & \vdots \\ \cdot & & & & O \\ & & & & O \\ O & & & O & a_{nn} \end{pmatrix}$$

$$x^{(k)} = \left(x_1^{(K)}, x_2^{(K)}, x_3^{(K)}, \ldots, x_n^{(k)} \right)^T \tag{1.4}$$

The method (1.3) adopts the additive splitting

$$\mathbf{A = D + M} \tag{1.5}$$

of **A**. The Gauss-Jacobi Method (GJM) in (1.3) takes the simple computational form, [8,9,10,11]

$$x_i^{(K+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{\substack{j=1 \\ j \neq i}}^{n} a_{ij} x_j^{(K)} \right); i = 1(1)n, k = 0, 1, 2, \ldots \tag{1.6}$$

written componentwise. This method is known to converge on the occasion that

- **A** is strictly diagonally dominant
- **A** is weakly diagonally dominant and irreducible
- $\{\rho(D^{-1}M), |D^{-1}M|\} < 1$, irrespective of the norm.
- **A** is positive definite L-matrix
- **A** is an L-matrix that is also an M-matrix.

Here $\rho(\mathbf{S})$ signifies the spectral radius of **S** [12,13,14,15,16].

2 MULTIPOINT ITERATION METHODS

The incorporation of past information in the GJM results in the 2-point iteration method

$$x_i^{(K+1)} = \frac{1}{a_{ii}}\left(b_i - \left\{ \sum_{\substack{j=1 \\ j\neq i}}^{m-1} a_{ij} x_j^{(K)} + \sum_{\substack{j=m \\ j\neq i}}^{n} a_{ij} x_j^{(K-1)} \right\} \right); i = 1(1)n, k = 1, 2,... \qquad (2.1)$$

This adopts the additive splitting

$$\mathbf{A = D + M_1 + M_2} \qquad (2.2)$$

and in matrix notation we shall have

$$X^{(k+1)} = D^{-1} b - D^{-1}M_1 x^{(k)} - D^{-1}M_2 x^{(K-1)} = C + B_1 x^{(k)} + B_2 X^{(k-1)}; K = 1,2 \qquad (2.3)$$

which when compared with the GJM, requires extra storage for the vector components of $\mathbf{x}^{(K-1)}$ in the past. This represents an extra cost in computation which may not be so significant for emerging parallel computers considering their enormous storage capacity. Here

$$M_1 = \begin{bmatrix} O & a_{12} & \cdots & a_{1m-1} & O & \cdots & O \\ a_{21} & O & & & & & \\ & & & a_{m-2,m-1} & O & & \\ & & & O & & & \\ & & & a_{m-2,m-1} & & & O \\ & & & & & & \\ a_{n1} & & \cdots & a_{n,m-1} & O & & O \end{bmatrix} \qquad (2.4)$$

$$M_2 = \begin{bmatrix} O & O & a_{1m} & \cdots & a_{1n} \\ & O & a_{m-1,m} & & a_{m-2,n} \\ & & O & & \\ & & a_{m+1,m} & & \\ & & & & a_{n,n-1} \\ O & O & a_{n,m} & a_{n,n-1} & O \end{bmatrix} \qquad (2.5)$$

Since the inverse of **A** exist, then

$$I = D^{-1}A - D^{-1}M_1 - D^{-1}M_2 \quad (2.6)$$

ensure the consistency of the splitting in (2.2). A variant of (2.1) we wish to present is the multipoint method

$$x_i^{(K+1)} = \frac{1}{a_{ii}}\left(b_i - \frac{1}{2}\left\{ \sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij}x_j^{(K)} + \sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij}x_j^{(k-1)} \right\} \right); i = 1(1)n, K = 1,2,... \quad (2.7)$$

This takes the equivalent form

$$\begin{aligned} x^{(K+1)} &= D^{-1}b - \frac{1}{2}[D^{-1}Mx^{(k)} + D^{-1}Mx^{(k-1)} \\ &= D^{-1}b - B[x^{(k)} + x^{(k-1)}] \quad ; \quad k = 1, 2 \end{aligned} \quad (2.8)$$

in matrix language which is in some sense an averaging method, with the definition that

$$x^{(K)} = \begin{cases} (o, o, ..., o)^T & ; \quad K = 0 \\ \left(\frac{b_1}{a_{11}}, \frac{b_2}{a_{22}}, ..., \frac{b_n}{a_{nn}} \right)^T & ; \quad K = 1 \end{cases} \quad (2.9)$$

provides ready made starting iterates.

3. CONVERGENCE ANALYSIS.

To study the convergence of the schemes (2.1) and (2.7) we define the block of solution

$$V_{K+1} = (x^{(K+1)}, x^{(K)})^T, \qquad V_K = (x^{(K)}, x^{(K-1)})^T, \; K = 1, 2, ... \quad (3.1)$$

and the block of constants

$$Q = (C, O)^T \; ; \; C = D^{-1}S \quad (3.2)$$

Then the methods (2.1) and (2.7) takes the new look

$$V_{K+1} = \begin{pmatrix} C \\ O \end{pmatrix} + \left\{ \begin{array}{l} \begin{pmatrix} B_1 & B_2 \\ I & O \end{pmatrix} V_K \,;\, B_s = -D^{-1} M_s, s = 1,2. \text{ for (2.1)} \\ \\ \begin{pmatrix} B & B \\ I & O \end{pmatrix} V_K \,;\, B = \frac{-D^{-1} M}{2}, K = 1,2,3,... \text{for (2.7)} \end{array} \right\} \quad (3.3)$$

respectively. In this formation, the multipoint method in (2.1) and (2.7) are now transformed into a stationary one-point iteration method in a higher dimensional space. The generalised form is given as

$$\mathbf{V}_{K+1} = \mathbf{Q} + \mathbf{E}\,\mathbf{V}_K,\ K = 1, 2, \ldots \quad (3.4)$$

where **E** can be identified quite trivially in our case of the methods (2.1) and (2.7) respectively. This block iteration method (3.4) possesses potential for parallelism since, if $\mathbf{V}_K$ is known the components of $\mathbf{V}_{K+1}$ can be computed in parallel. See Evans (1982). Now, from (3.4)

$$V_K = \left(I + \sum_{j=1}^{K-1} E^j \right) Q + E^K V_0 \,;\, K \geq 2 \quad (3.5)$$

Thus

$$V_K = (I - E)^{-1} (I - E^K)\, Q + E^K V_0 \quad (3.6)$$

Therefore, if

$$\lim_{K \to \infty} E^K V_0 = 0 \quad (3.7)$$

then

$$\{\| E\|, \rho(E)\} < 1 \quad (3.8)$$

irrespective of the arbitrariness in the choice of $\mathbf{V}_0$. Conclusively,

$$\lim_{K\to\infty} V_K = (I-E)^{-1}Q = \left\{ \begin{array}{l} \begin{pmatrix} I-B_1 & -B_2 \\ -I & I \end{pmatrix}^{-1} \begin{pmatrix} D^{-1}b \\ O \end{pmatrix}; \ (2.1) \\ \\ \begin{pmatrix} I-B & -B \\ -I & I \end{pmatrix}^{-1} \begin{pmatrix} D^{-1}b \\ O \end{pmatrix}; \ (2.7) \end{array} \right\} \qquad (3.9)$$

where in our case

$$E = \left\{ \begin{array}{l} \begin{pmatrix} B_1 B_2 \\ I \ O \end{pmatrix}; \ (2.1) \\ \\ \begin{pmatrix} B-B \\ I \ O \end{pmatrix}; \ (2.7) \end{array} \right\} = \left\{ \begin{array}{l} \begin{pmatrix} -D^{-1}M_1 & -D^{-1}M_2 \\ I & O \end{pmatrix}; \ (2.1) \\ \\ \begin{pmatrix} \frac{-D^{-1}M}{2} & \frac{-D^{-1}M}{2} \\ I & 0 \end{pmatrix}; \ (2.7) \end{array} \right\} \qquad (3.10)$$

respectively. We now set out to show in a limiting sense in (3.9), we shall have

$$\lim_{K\to\infty} V_k = \begin{pmatrix} A^{-1}b \\ \\ A^{-1}b \end{pmatrix} \qquad (3.11)$$

in both cases of the methods. Here, the need is to show that the left hand side of (3.9) reduces to that of (3.11). To achieve this we propose to prove the lemma that follows, whose application is needed to establish our case.

Lemma 3.1

Let **R - S** be a nonsingular matrix. Then the inverse of

$$G = \begin{pmatrix} R & S \\ -I & I \end{pmatrix} \qquad (3.12)$$

is given by

$$G^{-1} = \begin{pmatrix} (R-S)^{-1} & (R-S)^{-1}S \\ (R-S)^{-1} & (R-S)^{-1}R \end{pmatrix} \tag{3.13}$$

Proof:
Define that

$$f(\lambda) = \det(G - \lambda I) \tag{3.14}$$

By Cayley- Hamilton's theorem

$$(G) = O$$

That is

$$f(\lambda) = \det\begin{pmatrix} R-\lambda I & -S \\ -I & I-\lambda I \end{pmatrix} = \det\left[\lambda^2 I - \lambda(I+R) + (R-S)\right] = 0 \tag{3.15}$$

This implies that

$$G^2 - G\begin{pmatrix} I+R & 0 \\ 0 & I+R \end{pmatrix} + \begin{pmatrix} R-S & 0 \\ 0 & R-S \end{pmatrix} = O \tag{3.16}$$

The inverse G^{-1} exist since it is assumed that $(R-S)^{-1}$ exist and is therefore given as

$$G^{-1} = \begin{pmatrix} R-S & O \\ O & R-S \end{pmatrix}^{-1} \cdot \begin{pmatrix} I & S \\ I & R \end{pmatrix} \tag{3.17}$$

and we are through because

$$R(R-S)^{-1}S = S(R-S)^{-1}R$$

Back to the problem before us. In the case of the method (2.1) and (2.7) we identify **R** and **S** as

$$\left\{\begin{array}{l} R = I - B_1 = I + D^{-1}M_1,\ S = -B_2 = D^{-1}M_2;\ (2.1) \\ \\ R = I - B = I + \dfrac{D^{-1}}{2}, S = -B = \dfrac{D^{-1}M}{2};\ (2.7) \end{array}\right\} \tag{3.18}$$

from (3.9) respectively. Using the Lemma 3.1, we compute the inverse in (3.9) as

$$\begin{pmatrix} I - B_1 & B_2 \\ -I & I \end{pmatrix}^{-1} = \begin{pmatrix} (I - B_1 - B_2)^{-1} & (I - B_1 - B_2)^{-1}B_2 \\ (I - B_1 - B_2)^{-1} & (I - B_1 - B_2)^{-1}(I - B_1) \end{pmatrix} = T_1$$

$$\begin{pmatrix} I - B & -B \\ -I & I \end{pmatrix}^{-1} = \begin{pmatrix} (I - 2B)^{-1} & (I - 2B)^{-1}B \\ (I - 2B)^{-1} & (I - 2B)^{-1}(I - B) \end{pmatrix} = T_2 \tag{3.19}$$

Employing the definitions of (3.18), it is trivial to see that

$$\lim_{k\to\infty} V_k = \left\{\begin{array}{ll} T_1\begin{pmatrix} D^{-1}b \\ O \end{pmatrix}; & \text{for } (2.1) \\ T_2\begin{pmatrix} D^{-1}b \\ O \end{pmatrix}; & \text{for } (2.7) \end{array}\right\} = \begin{pmatrix} A^{-1}b \\ A^{-1}b \end{pmatrix} \tag{3.20}$$

in both methods. This shows that the methods (2.1) and (2.7) converges on the condition of (3.7). We present further convergence conditions for the iterative methods (2.1) and (2.7).

Lemma 3.2

Let the coefficient matrix **A** be strictly diagonally dominant, that is

$$\sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}| < |a_{ii}|;\ i = 1(1)n \tag{3.21}$$

Then the iterative methods (2.1) and (2.7) are convergent.

Proof:

Define the error incured at the kth iteration as

$$e^{(K)} = x - x^{(K)} = \left\{ \begin{array}{l} (x_1 - x_1^{(K)}, x_2 - x_2^{(K)}, \ldots, x_n - x_n^{(K)})^T = \\ \\ (e_1^{(K)}, e_2^{(K)}, \ldots, e_n^{(K)})^T ; K = 1,2,3,\ldots \end{array} \right\} \tag{3.22}$$

Thus

$$|e_i^{(K+1)}| \leq \frac{1}{|a_{ii}|} \left\{ \begin{array}{l} \left[\sum_{\substack{j=1 \\ j \neq i}}^{m-1} |a_{ij}| \, |e_j^{(K)}| + \sum_{\substack{j=m \\ j \neq i}}^{n} |a_{ij}| \, |e_j^{(K-1)}| \right]; (2.1) \\ \\ \frac{1}{2} \left[\sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{ij}| \, |e_j^{(K)}| + \sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{ij}| \, |e_j^{(K-1)}| \right]; (2.7) \end{array} \right\} \tag{3.23}$$

To simplify analysis, define that

$$\gamma_1 = \max_{1 \leq i \leq n} \left\{ \sum_{\substack{j=1 \\ j \neq i}}^{m-1} \left| \frac{a_{ij}}{a_{ii}} \right| \right\}, \gamma_2 = \max_{1 \leq i \leq n} \left\{ \sum_{\substack{j=m \\ j \neq i}}^{n} \left| \frac{a_{ij}}{a_{ii}} \right| \right\} \tag{3.24}$$

$$\gamma = \max_{1 \leq i \leq n} \left\{ \sum_{\substack{j=1 \\ j \neq i}}^{n} \left| \frac{a_{ij}}{a_{ii}} \right| \right\}$$

Then (3.23) reduces nicely to

$$\left|e^{(K+1)}\right| \leq \left\{ \begin{array}{l} \gamma_1 \left|e^{(K)}\right| + \gamma_2 \left|e^{(K-1)}\right| ; \ (2.1) \\ \gamma_2 \left[\left|e^{(K)}\right| + \left|e^{(K-1)}\right| \right] ; (2.7) \end{array} \right\} \tag{3.25}$$

respectively. By further simplification

$$\left|e^{(K+1)}\right| \leq \left\{ \begin{array}{l} (\gamma_1 + \gamma_2)^K \max\left\{\left|e^{(o)}\right|, \left|e^{(1)}\right|\right\} ; (2.1) \\ \gamma^k \max\left\{\left|e^{(o)}\right|, \left|e^{(1)}\right|\right\} ; \quad (2.7) \end{array} \right\} \tag{3.26}$$

It is reasonable that if **A** is strictly diagonally dominant then

$$\gamma_1 + \gamma_2 = \gamma = \max_i \left\{ \sum_{\substack{j=1 \\ j \neq i}}^{n} \left| \frac{a_{ij}}{a_{ii}} \right| \right\} < 1, i = 1(1)n \tag{3.27}$$

on which condition convergence of the methods is assured. Readily, we may write that

$$| e^{(K+1)} |_\infty \leq | D^{-1} M |_\infty^K \max_{S=0,1} \{ | e^{(S)} |_\infty \} ; | D^{-1} M |_\infty < 1 \tag{3.28}$$

4 ROUNDOFF PROPAGATION

In general, roundoff error incurred in an iteration, process can be much of a serious problem especially the linear system is ill-conditioned [10,7] and more so when the precision employed in the computing process is far from being worthwhile. If the computation in (2.1) is performed in p-digit floating-point arithmetic, the roundoff errors propagate their effects in the case of the method (2.1), as follows:

$$x_r^{(K+1)} = \frac{-1}{a_{rr}} [[\sum_{j=T_{1,r}}^{T_{p,r}} a_{rj} x_r^{(K)} (1+\lambda_j) \prod_{s=m(j)}^{P} (1+\eta_s)$$

$$+ \sum_{j=T_{p+1,r}}^{T_{q,r}} a_{rj} x_j^{(K-1)} (1+\lambda_j) \prod_{s=m'(j)}^{q-p} (1+\eta_s)] (1+\delta_1) - b_r] (1+\delta_2)(1+\delta_3) \tag{4.1}$$

In a sense of Ralston et al (1983) the case of GJM takes the form

$$x_r^{(K+1)} = \frac{-1}{a_{rr}}\left[\sum_{j=T_{i,r}}^{T_{p,r}} a_{rj}\, x_r^{(K)}(1+\lambda_j)\prod_{s=m(j)}^{P}(1+\eta_s) - b_r\right](1+\delta_1)(1+\delta_3) \quad (4.2)$$

Similar expression exist for the method (2.7), but in (4.1)

$$1 \le T_{1,r} < \ldots < T_{p,r} \le m-1 \le m \le T_{p+1,r} < \ldots, T_{q,r} \le n$$

and

$$m(j) = \left\{\begin{array}{ll} 2 \;; & j = T_{1,r} \\ 1 \;; & j = T_{t,r};\ t>1 \end{array}\right\},\quad m'(j) = \left\{\begin{array}{ll} 2 \;; & j = T_{p+1,r} \\ t-p \;; & j = T_{t,r},\ t > p+1 \end{array}\right\} \quad (4.3)$$

Furthermore,

λ_j: relative error in the multiplication of a_{rj} by $x_j^{(K)}$ in (2.1) $\prod_s (1+\eta_s)$: relative error in the first accumulated sum in (2.1)

δ_1: relative error in the addition of the two sums of the right of (2.1)

δ_2, δ_3: relative errors in the addition of b_r and division by a_{rr} respectively in (2.1)

Expressing the result of (4.1) in matrix notations, we have

$$\tilde{D}\, x^{(K+1)} = -(\tilde{M}_1 x^{(K)} + \tilde{M}_2 x^{(K-1)} - b)$$

where

$$\tilde{d}_{rr} = \frac{a_{rr}}{(1+\delta_2)(1+\delta_3)};\ r = 1(1)n$$

$$\tilde{M}_{rj}^{(1)} = \left\{\begin{array}{ll} a_{rj}(1+\lambda_j)(1+\delta_1)\prod_s (1+\eta_s) \;; & r \ne j,\ j = 1(1)m-1 \\ 0 \;; & r = j \end{array}\right\}$$

$$\tilde{M}_{rj}^{(2)} = \left\{\begin{array}{ll} a_{rj}(1+\lambda_j)(1+\delta_1)\prod_s (1+\eta_s) \;; & r \ne j,\ j = 1(1)n \\ 0 \;; & r = j \end{array}\right\}$$

are the entries of $\tilde{D}, \tilde{M}_1$ and $\tilde{M}_2$ respectively. Similar analysis exist for (2.7) and the GJM, from which it is deductive that roundoff effects are more influential in the methods (2.1) and (2.7) than in the GJM. However, if the precision is reasonably high and such that the coefficient matrix **A** is well-conditioned then the effects are sure to die off in the long run on the iteration index k when the conditions of (3.21) or (3.7) holds as would be demanded.

5 CONCLUSION

Typically, the methods can be written as

$$\begin{pmatrix} x^{(K+1)} \\ x^{(K)} \end{pmatrix} = \left\{ \begin{array}{l} \begin{pmatrix} -D^{-1}M_1 & -D^{-1}M_2 \\ I & 0 \end{pmatrix} \begin{pmatrix} x^{(K)} \\ x^{(K-1)} \end{pmatrix} + \begin{pmatrix} D^{-1}b \\ O \end{pmatrix}; \quad (2.1) \\ \\ \begin{pmatrix} -D^{-1}M & -D^{-1}M \\ I & 0 \end{pmatrix} \begin{pmatrix} x^{(K)} \\ x^{(K-1)} \end{pmatrix} + \begin{pmatrix} D^{-1}b \\ O \end{pmatrix}; \quad (2.7) \end{array} \right\} \tag{5.1}$$

according as the method in use. This points to the fact that these iteration processes are solving the system of linear equations

$$\left(\begin{array}{c|c} A & O \\ \hline O & I \end{array} \right) \left(\begin{array}{c} x \\ \hline x \end{array} \right) = \left(\begin{array}{c} b \\ \hline x \end{array} \right) \tag{5.2}$$

The second row is a dummy system of equations which arises from the need to reserve storage locations for the backward components $\mathbf{x}^{(K-1)}$ in (5.1). The extra cost of doing this places the GJM and GSM at an advantage despite that the convergence rate of all these methods is linear. From the numerical results we provide, the accuracy of the multipoint methods (2.1) and (2.7) is not better than that of GJM and GSM. A reason for this handicap can be attributed in part to the fact that the effects of roundoffs are far more significant in multipoint methods than that of the GJM. This however, raises the question as to whether there is any need pursuing multipoint iteration methods for sparse linear system of equations as a separate method. In Ikhile (1999) we reconsidered this in a far more general formalism to conclude that multipoint methods may well not be worth the efforts after all because of their extra computational cost arising from the need for a book-keeping exercise which increases with increasing number of the blocks of past solution. This inefficiency of multipoint methods as a separate method becomes more of a disadvantage when it is discovered that their convergence rate is not

better than that of GSM especially for strictly diagonally dominant coefficient matrix. Conclusively, multipoint iteration methods are no doubt of theoretical interest.

6. NUMERICAL EXPERIMENT

We consider the numerical solution of the diagonally dominant linear system

Test I

$$A = \begin{pmatrix} 4 & -2 & 0 & 0 \\ -1 & 5 & -1 & 0 \\ 0 & -1 & 4 & 2 \\ 0 & 0 & 2 & 3 \end{pmatrix}, b = \begin{pmatrix} 0 \\ 2 \\ 3 \\ 2 \end{pmatrix}$$

Test II

$$A = \begin{pmatrix} 4 & -1 & 0 & -1 & 0 & 0 \\ -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & -1 & 4 & 0 & 0 & -1 \\ -1 & 0 & 0 & 4 & -1 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 \\ 0 & 0 & -1 & 0 & -1 & 4 \end{pmatrix}, b = \begin{pmatrix} 0 \\ 5 \\ 0 \\ 6 \\ -2 \\ 6 \end{pmatrix}$$

The error incurred using the methods (2.1) and (2.7) and the GJM are compared after a few iterations in Table 6.1 Table 6.2 and Table 6.3 respectively.

We have used the abbreviations

GJM: Gauss-Jacobi Method for (1.6)

EGJM(I): Extended GJM for (2.1)

EGJM (II): Extended GJM for (2.7)

GSM: Gauss-Seidel Method for easy reference.

TABLE 6.1

TEST I
TABLE OF ERRORS

ITERATIVE INDEX	GJM	GSM	EGJM (I)	EGJM(II)
8	0.034089	0.002517811	0.165085	0.339363103
9	0.022726	0.00107856	0.1154195	0.688181139
10	0.0145705	0.000461305	0.093364083	0.690804125
11	0.009713666	0.000197191	0.057597083	0.018682017
12	0.00622	0.000084375	0.043184016	0.234453839

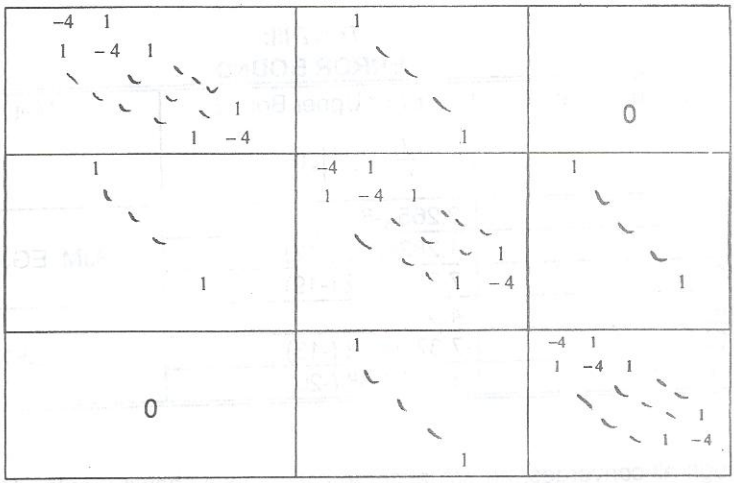
TABLE 6.2

TEST II

ITERATIVE INDEX	GJM	GSM	EGJM (I)	EGJM(II)
8	0.049179077	0.00079806	0.280273438	0.0362088
9	0.02968216	0.0000605176	0.161865234	0.0254934
10	0.017914772	0.0001057460	0.11462424	0.0290628
11	0.01081252	0.000038734	0.09753418	0.0159097
12	0.006525934	0.000058424	0.090545655	0.0119036

For problem I and II it is easy to observe from the Table 6.1 and Table 6.2 that the Gauss-Seidel method is fastest in convergence.

TEST III

$A =$ 

where **b** is normalised in the sense that $b_i = \sum_{j=1}^{n} a_{ij}; i = 1,2,3,...$ to have the solution as $x = (1, 1, 1, ..., 1)^T$.

Here

$$\left|e^{(K)}\right| \leq \begin{cases} \left(\frac{3}{4}\right)^K \left|e^{(0)}\right|; \text{GJM}; K = 0,1,2,... \\ \left(\frac{2}{3}\right)^K \left|e^{(0)}\right|; \text{GSM} \\ \left(\frac{3}{4}\right)^K \max_{S=0,1} \left\{\left|e^{(S)}\right|\right\}; \text{EGJM(I, II)}, \end{cases}$$

TABLE 6.3

TEST III:
ERROR BOUND

Iteration Index: K	Error Upper Bound: $\frac{\mu^K}{1-\mu}\left\|e^{(0)}\right\|, 0<\mu<1$	Method
50	2.265 (-6)	GJM, EGJM (I, II)
100	1.2828808 (-12)	
150	7.2652322 (-19)	
50	4.0 (-9)	GSM
100	7.3789632 (-18)	
150	1.1572638 (-26)	

Although all converges, an actual computation have shown that the GSM is fastest in convergence. In fact, the actual rating is as follows:

$N_{EGJM(I)} > N_{EGJM(II)} > N_{GJM} > N_{GSM}$

This is in terms of the number of iterations $N_{(\ )}$ to a given error tolerance.

REFERENCES

[1] Burden R.L., Faires J.D. and Reynold A.C. ; (1981): Numerical Analysis. Prindle, Weber and Schmidt, Boston, USA.

[2] David, R.H.; *Experiments in Computational Matrix Algebra.* The Random House/Birkhauser Mathematics Series, (1988).

[3] Eikemo, M.S., Espedal, M.S. and Fladmark, G.; On the Numerical Solution of a Three Dimensional Extrusion Model. Computing and Visualisation in Science, Vol. 1, pp. 1 - 14, Springer-Verlag, (1997).

[4] Evans J.D.: Parallel Numerical Algorithms for Linear Systems. Parallel Processing Systems (ed D. Evans). Cambridge University Press, (1982).

[5] Gander, W.; and Hrebicek, J.; Solving Problems in Scientific Computing Using MAPLE and MATLAB. Springer-Verlag, (1998).

[6] Hackbusch, W. and Sauter, S.A., Composite Finite Elements for Problems Containing Small Geometric Details. part II: Implementation and Numerical Results. Computing and Visualisation in Science, Vol.1, pp. 15-25, Springer-Verlag, (1997).

[7] Ikhile, M.N.O: Multipoint Methods for large sparse system of linear Equations. Submitted to Journal of Computer Mathematics U.K (1999).

di J.J.: Parallel Algorithms and Matrix Computation. Oxford Applied thematics and Computing Science Series. Clarendon Press, Oxford, 88).

theij R.M.M.: Direct Solution of certain sparse Linear System. mputational Ordinary Differential Equations (ed. S.O. Fatunla), 1992.

ston, A. and Rabinowitz, P.; A First Course in Numerical Analysis. graw-Hill Intern. Book Company, (1983).

d, J.K.; Large Sparse Sets of Linear Equations. Proceedings of the ford Conference of the Institute of Mathematics and its Applications, ademic Press (1971).

n Norton, R.; The Solution of Linear Equations by the Gauss-Seidel thod. Mathematical Methods for Digital Computers, (ed. Ralston, A. and f, H.S.) John Wiley and Sons, Inc. New York, (1967).

warz, H.R.; Numerical Analysis: A Comprehensive Introduction. John Wiley Son Ltd, (1989).

awrz, H.R., Rutishauser, H. and Stiefel, E.; Numerical Analysis of mmetric Matrices. Prentice-Hall, Inc. Englewood Cliffs, N.J., (1973).

ldon, J.W.; Iterative Methods for the Solution of Elliptic Partial erential Equations. Mathematical Methods for Digital Computers (ed. ston, A and Wilf, H.S.). John Wiley and Sons Inc., (1971).

ng M.D. and Gregory T. R. ; A Survey of Numerical Mathematics: Vol. I II. Addison-Wesley Publishing Comp. Inc. (1972).