International Journal of Mathematics and Statistics Invention (IJMSI)
E-ISSN: 2321 – 4767 P-ISSN: 2321 - 4759
www.ijmsi.org Volume 4 Issue 8 || October. 2016 || PP-20-26

Solving Poisson's Equation Using Preconditioned Nine-Point Group SOR Iterative Method

Abdulkafi Mohammed Saeed [1]
Department of Mathematics, College of Science, Qassim University, Saudi Arabia

ABSTRACT*: A well-designed preconditioning of the partial differential equations problems reduces the number of iterations to reach convergence. Dramatic improvements are possible, but the difficulty is to construct the suitable preconditioner.The construction of a specific splitting-type preconditioner in block formulation for a class of group relaxation iterative methods derived from the finite difference approximations have been shown to improve the convergence rates of these methods. This paper is concerned with the application of suitable preconditioning techniques to the Nine-Point Group SOR (N-P SOR) iterative method for solving Poisson's Equation. Preconditioning strategies which improve the rate of convergence of these iterative methods are investigated. The results reveal the improvements on the convergence rate and the efficiency of the proposed preconditioned Group iterative method.*

KEYWORDS - *Poisson's Equation, Preconditioned Nine-Point Group SOR Iterative Method.*

I. INTRODUCTION

Elliptic equations describe problems in a closed region. The Poisson and Laplace equations are examples of the elliptic equations. Only boundary conditions are considered because the dependent variable does not depend on time. In this work, we will consider Poisson equation in the form:

$$\nabla^2 u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y), \quad (x, y) \in \Omega \tag{1.1}$$

with specific Dirichlet boundary conditions

$$U(x, y) = g(x, y), \qquad (x, y) \in \partial\Omega .$$

The well-known classical iterative methods are Jacobi, Gauss-Seidel and Successive Over Relaxation (SOR) methods. The Jacobi and Gauss-Seidel methods could be used for such systems with some success, not so much because of the reduction in the computational work, but mainly because of the limited amount of memory that is required.

In SOR method, we have to determine the parameter *w*, where a suitable value of *w* could lead to drastic improvements in convergence. Because of that, the SOR method became very popular and this method was selected as the method in computer codes, in order to solve large practical problems such as weather prediction and nuclear reactor diffusion [1].

It has been affirmed that the discretisation of PDEs using finite difference schemes normally yield a system of linear equations, which are large and sparse in nature. Iterative methods are usually used to solve these types of systems since these methods need less storage and are capable of preserving the sparsity property of the large system. Many problems in various fields, such as engineering, science and the quantitative study of business and economic problems associate with the linear systems of equations . In Yousif and Evans [2], the explicit group over relaxation methods for the numerical solution of the sparse linear systems has been presented. The group method has been shown computationally superior in comparison with the implicit one-line and two-line block SOR iterative method.

Suppose (1.1) is discretised using some finite difference scheme, this will normally lead to a large, block and sparse system of equations. Many researchers have considered preconditioners, applied to linear systems ([3], [4], [5], [6]).

In Lee [7], preconditioners had been successfully applied on the standard five point formula in solving the Poisson problem with Dirichlet boundary condition and the numerical experiments yield some encouraging results.

Equation (1.1) may be approximated at the point (x_i, y_j) in many ways. Assume that a rectangular grid in the (x,y) plane with equal grid spacing h in both directions with $x_i = ih, y = jh(i, j = 0,1,....N)$ are used, where $u_{i,j} = u(x_i, y_j)$ and $h = 1/N$. By neglecting terms of $O(h^2)$, we obtain the simplest approximation for (1.1) which is known as the standard five-point difference formula:

$$u_{i,j+1} + u_{i,j-1} + u_{i+1,j} + u_{i-1,j} - 4u_{ij} = h^2 f_{ij} \qquad (1.2)$$

The objective of this research is to find the most efficient group SOR iterative method for solving elliptic partial differential equations. We will compare the point, the four-point group and the nine-point group SOR iterative method in solving the two dimensional Poisson equation.

The outline of this paper is as follows: An overview of the formulation of the Nine-Point Group SOR (NPSOR) iterative method for solving Poisson's Equation will be given in Section 2. We proposed a new type of preconditioned formula and apply it to the N-P SOR method for solving Poisson's Equation in Section 3. The numerical results are presented to show the efficiency of the preconditioned N-P SOR method in Section 4 and the concluding remarks is given in Section 5.

II. FORMULATION OF THE NINE-POINT GROUP ITERATIVE METHOD

In the group 9-point method the mesh points are grouped together in blocks of nine. The points involved in updating u_{ij} are also using standard five-point formula. The solution domain is divided into groups of nine points as shown in Fig.1:

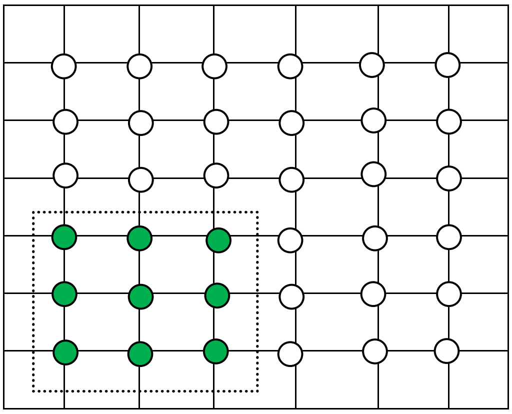

Fig. 1: The solution domain for group of nine points

In matrix notation, the system of nine equations can be written as:

$$\begin{bmatrix} 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 4 & 0 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 4 & -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 4 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 \end{bmatrix} \begin{bmatrix} u_{ij} \\ u_{i+1,j} \\ u_{i+2,j} \\ u_{i,j+1} \\ u_{i+1,j+1} \\ u_{i+2,j+1} \\ u_{i,j+2} \\ u_{i+1,j+2} \\ u_{i+2,j+2} \end{bmatrix} = \begin{bmatrix} u_{i-1,j} + u_{i,j-1} - h^2 f_{ij} \\ u_{i+1,j-1} - h^2 f_{i+1,j} \\ u_{i+2,j-1} + u_{i+3,j} - h^2 f_{i+2,j} \\ u_{i-1,j+1} - h^2 f_{i,j+1} \\ -h^2 f_{i+1,j+1} \\ u_{i+3,j+1} - h^2 f_{i+2,j+1} \\ u_{i-1,j+2} + u_{i,j+3} - h^2 f_{i,j+2} \\ u_{i+1,j+3} - h^2 f_{i+1,j+2} \\ u_{i+3,j+2} + u_{i+2,j+3} - h^2 f_{i+2,j+2} \end{bmatrix} \qquad (2.1)$$

The inverse form of the above system is:

$$\begin{bmatrix} u_{ij} \\ u_{i+1,j} \\ u_{i+2,j} \\ u_{i,j+1} \\ u_{i+1,j+1} \\ u_{i+2,j+1} \\ u_{i,j+2} \\ u_{i+1,j+2} \\ u_{i+2,j+2} \end{bmatrix} = \begin{bmatrix} \frac{67}{224} & \frac{22}{224} & \frac{7}{224} & \frac{22}{224} & \frac{14}{224} & \frac{22}{224} & \frac{22}{224} & \frac{22}{224} & \frac{22}{224} \\ \frac{11}{112} & \frac{37}{112} & \frac{11}{112} & \frac{7}{112} & \frac{14}{112} & \frac{7}{112} & \frac{3}{112} & \frac{5}{112} & \frac{3}{112} \\ \frac{7}{224} & \frac{22}{224} & \frac{67}{224} & \frac{6}{224} & \frac{14}{224} & \frac{22}{224} & \frac{3}{224} & \frac{6}{224} & \frac{7}{224} \\ \frac{11}{112} & \frac{7}{112} & \frac{3}{112} & \frac{37}{112} & \frac{14}{112} & \frac{5}{112} & \frac{11}{112} & \frac{7}{112} & \frac{3}{112} \\ \frac{1}{16} & \frac{2}{16} & \frac{1}{16} & \frac{2}{16} & \frac{6}{16} & \frac{2}{16} & \frac{1}{16} & \frac{2}{16} & \frac{1}{16} \\ \frac{3}{112} & \frac{7}{112} & \frac{11}{112} & \frac{5}{112} & \frac{14}{112} & \frac{37}{112} & \frac{3}{112} & \frac{7}{112} & \frac{11}{112} \\ \frac{7}{224} & \frac{6}{224} & \frac{3}{224} & \frac{22}{224} & \frac{14}{224} & \frac{6}{224} & \frac{67}{224} & \frac{22}{224} & \frac{7}{224} \\ \frac{3}{112} & \frac{5}{112} & \frac{3}{112} & \frac{7}{112} & \frac{14}{112} & \frac{7}{112} & \frac{11}{112} & \frac{37}{112} & \frac{11}{112} \\ \frac{3}{224} & \frac{6}{224} & \frac{7}{224} & \frac{6}{224} & \frac{14}{224} & \frac{22}{224} & \frac{7}{224} & \frac{22}{224} & \frac{67}{224} \end{bmatrix} \begin{bmatrix} u_{i-1,j} + u_{i,j-1} - h^2 f_{ij} \\ u_{i+1,j-1} - h^2 f_{i+1,j} \\ u_{i+2,j-1} + u_{i+3,j} - h^2 f_{i+2,j} \\ u_{i-1,j+1} - h^2 f_{i,j+1} \\ -h^2 f_{i+1,j+1} \\ u_{i+3,j+1} - h^2 f_{i+2,j+1} \\ u_{i-1,j+2} + u_{i,j+3} - h^2 f_{i,j+2} \\ u_{i+1,j+3} - h^2 f_{i+1,j+2} \\ u_{i+3,j+2} + u_{i+2,j+3} - h^2 f_{i+2,j+2} \end{bmatrix}$$

Hence, the explicit 9-point group iterative equations are given by:

$$\begin{aligned}
u_{ij} &= \frac{1}{224}[67t_1 + 22t_2 + 7t_7 - 14t_0 + 6t_5 + 3t_6], & u_{i+1,j} &= \frac{1}{112}[37t_{19} + 11t_8 + 7t_9 - 14t_0 + 5t_{20} + 3t_{10}], \\
u_{i+2,j} &= \frac{1}{224}[67t_3 + 22t_{13} + 7t_{18} - 14t_0 + 6t_{14} + 3t_4], & u_{i,j+1} &= \frac{1}{112}[37t_{21} + 11t_{15} + 7t_{16} - 14t_0 + 5t_{22} + 3t_{17}], \\
u_{i+1,j+1} &= \frac{1}{16}[2t_{11} - 6t_0 + t_{12}], & u_{i+2,j+1} &= \frac{1}{112}[37t_{22} + 11t_{17} + 7t_{16} - 14t_0 + 5t_{21} + 3t_{15}], \\
u_{i,j+2} &= \frac{1}{224}[67t_4 + 22t_{14} + 7t_{18} - 14t_0 + 6t_{13} + 3t_3], & u_{i+1,j+2} &= \frac{1}{112}[37t_{20} + 11t_{10} + 7t_9 - 14t_0 + 5t_{19} + 3t_8], \\
u_{i+2,j+2} &= \frac{1}{224}[67t_6 + 22t_5 + 7t_7 - 14t_0 + 6t_2 + 3t_1] & &
\end{aligned} \tag{2.2}$$

where:

$$\begin{aligned}
& t_0 = h^2 f_{i+1,j+1}, \quad t_1 = u_{i-1,j} + u_{i,j-1} - h^2 f_{i+1,j+1}, \quad t_2 = u_{i+1,j-1} + u_{i-1,j+1} - h^2 f_{i+1,j} - h^2 f_{i,j+1}, \\
& t_3 = u_{i+2,j-1} + u_{i+3,j} - h^2 f_{i+2,j}, \quad t_4 = u_{i-1,j+2} + u_{i,j+3} - h^2 f_{i,j+2}, \quad t_5 = u_{i+3,j+1} + u_{i+1,j+3} - h^2 f_{i+2,j+1} - h^2 f_{i+1,j+2}, \\
& t_6 = u_{i+3,j+2} + u_{i+2,j+3} - h^2 f_{i+2,j+2}, \quad t_7 = t_3 + t_4, \quad t_8 = t_1 + t_3, \quad t_9 = u_{i+3,j+1} + u_{i-1,j+1} - h^2 f_{i+2,j+1} - h^2 f_{i,j+1}, \\
& t_{10} = t_4 + t_6, \quad t_{11} = t_2 + t_5, \quad t_{12} = t_8 + t_{10}, \quad t_{13} = u_{i+1,j-1} + u_{i+3,j+1} - h^2 f_{i+1,j} - h^2 f_{i+2,j+1}, \\
& t_{14} = u_{i-1,j+1} + u_{i+1,j+3} - h^2 f_{i,j+1} - h^2 f_{i+1,j+2}, \quad t_{15} = t_1 + t_4, \quad t_{16} = u_{i+1,j-1} + u_{i+1,j+3} - h^2 f_{i+1,j} - h^2 f_{i+1,j+2}, \\
& t_{17} = t_3 + t_6, \quad t_{18} = t_1 + t_6, \quad t_{19} = u_{i+1,j-1} - h^2 f_{i+1,j}, \quad t_{20} = u_{i+1,j+3} - h^2 f_{i+1,j+2}, \quad t_{21} = u_{i-1,j+1} - h^2 f_{i,j+1}, \\
& t_{22} = u_{i+3,j+1} - h^2 f_{i+2,j+1}
\end{aligned}$$

This method proceeds with iterative evaluation of solution in groups of nine points throughout the entire solution domain using all nine equations (2.2). The process is continuous until convergence is achieved. The nine-point group can be represented by the computational molecule shown in Fig. 2.

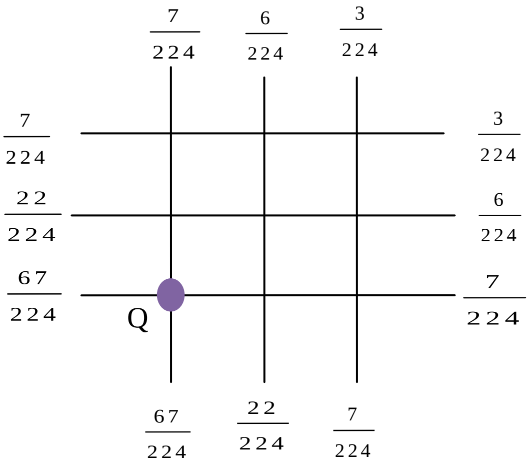

Fig. 2: The computational molecule for the nine- point group at point Q

III. DERIVATION OF THE PROPOSED PRECONDITIONED N-P SOR ITERATIVE METHOD

According to equation (1.2) in section 3.2, the SOR iterative scheme can be written as

$$u_{ij}^{(k+1)} = \frac{\omega}{4}\left(u_{i+1,j}^{(k)} + u_{i,j+1}^{(k)} + u_{i-1,j}^{(k+1)} + u_{i,j-1}^{(k+1)} - h^2 f_{ij}\right) + (1-\omega)u_{ij}^{(k)}$$

Now, from (2.2), we can build the nine-point SOR iterative scheme as follows:

$$\begin{aligned}
u_{ij}^{(k+1)} &= \frac{1}{224}[\omega(67t_1 + 22t_2 + 7t_7 - 14t_0 + 6t_5 + 3t_6)] + (1-\omega)u_{ij}^{(k)},\\
u_{i+1,j}^{(k+1)} &= \frac{1}{112}[\omega(37t_{19} + 11t_8 + 7t_9 - 14t_0 + 5t_{20} + 3t_{10})] + (1-\omega)u_{i+1,j}^{(k)},\\
u_{i+2,j}^{(k+1)} &= \frac{1}{224}[\omega(67t_3 + 22t_{13} + 7t_{18} - 14t_0 + 6t_{14} + 3t_4)] + (1-\omega)u_{i+2,j}^{(k)},\\
u_{i,j+1}^{(k+1)} &= \frac{1}{112}[\omega(37t_{21} + 11t_{15} + 7t_{16} - 14t_0 + 5t_{22} + 3t_{17})] + (1-\omega)u_{i,j+1}^{(k)},\\
u_{i+1,j+1}^{(k+1)} &= \frac{1}{16}[\omega(2t_{11} - 6t_0 + t_{12})] + (1-\omega)u_{i+1,j+1}^{(k)},\\
u_{i+2,j+1}^{(k+1)} &= \frac{1}{112}[\omega(37t_{22} + 11t_{17} + 7t_{16} - 14t_0 + 5t_{21} + 3t_{15})] + (1-\omega)u_{i+2,j+1}^{(k)},\\
u_{i,j+2}^{(k+1)} &= \frac{1}{224}[\omega(67t_4 + 22t_{14} + 7t_{18} - 14t_0 + 6t_{13} + 3t_3)] + (1-\omega)u_{i,j+2}^{(k)},\\
u_{i+1,j+2}^{(k+1)} &= \frac{1}{112}[\omega(37t_{20} + 11t_{10} + 7t_9 - 14t_0 + 5t_{19} + 3t_8)] + (1-\omega)u_{i+1,j+2}^{(k)},\\
u_{i+2,j+2}^{(k+1)} &= \frac{1}{224}[\omega(67t_6 + 22t_5 + 7t_7 - 14t_0 + 6t_2 + 3t_1)] + (1-\omega)u_{i+2,j+2}^{(k)},
\end{aligned} \tag{3.1}$$

where

$$t_0 = h^2 f_{i+1,j+1}, \quad t_1 = u_{i-1,j}^{(k+1)} + u_{i,j-1}^{(k+1)} - h^2 f_{i,j}, \quad t_2 = u_{i+1,j-1}^{(k+1)} + u_{i-1,j+1}^{(k+1)} - h^2 f_{i+1,j} - h^2 f_{i,j+1},$$

$$t_3 = u_{i+2,j-1}^{(k+1)} + u_{i+3,j}^{(k)} - h^2 f_{i+3,j} - h^2 f_{i+2,j}, \qquad t_4 = u_{i-1,j+2}^{(k+1)} + u_{i,j+3}^{(k)} - h^2 f_{i,j+2},$$

$$t_5 = u_{i+3,j+1}^{(k)} + u_{i+1,j+3}^{(k)} - h^2 f_{i+2,j+1} - h^2 f_{i+1,j+2}, \qquad t_6 = u_{i+2,j+2}^{(k)} + u_{i+2,j+3}^{(k)} - h^2 f_{i+2,j+2},$$

$$t_7 = t_3 + t_4, \qquad t_8 = t_1 + t_3, \qquad t_9 = u_{i+3,j+1}^{(k)} + u_{i-1,j+1}^{(k+1)} - h^2 f_{i+2,j+1} - h^2 f_{i,j+1},$$

$$t_{10} = t_4 + t_6, \quad t_{11} = t_2 + t_5, \quad t_{12} = t_8 + t_{10}, \qquad t_{13} = u_{i+1,j-1}^{(k+1)} + u_{i+3,j+1}^{(k)} - h^2 f_{i+1,j} - h^2 f_{i+2,j+1},$$

$$t_{14} = u_{i-1,j+1}^{(k+1)} + u_{i+1,j+3}^{(k)} - h^2 f_{i,j+1} - h^2 f_{i+1,j+2}, \quad t_{15} = t_1 + t_4, \quad t_{16} = u_{i+1,j-1}^{(k+1)} + u_{i+1,j+3}^{(k)} - h^2 f_{i+1,j} - h^2 f_{i+1,j+2},$$

$$t_{17} = t_3 + t_6, \quad t_{18} = t_1 + t_6, \quad t_{19} = u_{i+1,j-1}^{(k+1)} - h^2 f_{i+1,j},$$

$$t_{20} = u_{i+1,j+3}^{(k)} - h^2 f_{i+1,j+2}, \quad t_{21} = u_{i-1,j+1}^{(k+1)} - h^2 f_{i,j+1}, \quad t_{22} = u_{i+3,j+1}^{(k)} - h^2 f_{i+2,j+1},$$

It is well known that the resulting system for applying N-P SOR for solving (1.1) can be written as:

$$A\bar{u} = \bar{f} \tag{3.2}$$

Matrix A is can be written as $A = D - L - U$, where D is a diagonal matrix and L and U are strictly lower and upper triangular matrices. The convergence rates of (3.2) depend on the spectral properties of the coefficient matrix A. A preconditioner is a matrix that transforms the linear system into one that is equivalent in the sense that it has the same solution, but that has more favourable spectral properties.

For the nine-point group method, the matrix A, vectors $\bar{u}$ and $\bar{f}$ are as defined in (2.1)). Therefore the precondetioner M is obtained in the form: $M = D - L$, and then, we can write the precontioned system as the following:

$$M(A)\bar{u} = M\bar{f} \Rightarrow (D - L)(A)\bar{u} = (D - L)\bar{f} \tag{3.3}$$

IV. NUMERICAL EXPERIMENTS AND RESULTS

In order to compare the standard five-point, nine-point group SOR and preconditioned nine-point group SOR iterative methods, some numerical experiments have been performed. These methods were implemented to model problem of Poisson equation in the form:

$$\nabla^2 u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = (x^2 + y^2)e^{xy}, \tag{4.1}$$

with $\quad u(x,0) = u(0,y) = 1, \quad u(x,1) = e^x, \quad u(1,y) = e^y, \qquad 0 \le x, y \le 1.$

The exact solution for this problem is $u(x,y) = e^{xy}$. In this experimental work, we choose the value of tolerance; $\varepsilon = 10^{-6}$. The computer processing unit is Intel(R) Core(TM) i5 with memory of 4Gb and the software used to implement and generate the results was Developer C++ Version 4.9.9.2. We have computed the average absolute errors and record the number of iterations for convergence for different size of grids 45, 85, 105, 145, 185 and 225.

Table 1: Comparison of number of iterations, execution time and over relaxation parameter for standard five-point, N-P SOR and preconditioned N-P SOR iterative methods

N	Standard Five-Point				N-P Group SOR				Preconditioned N-P Group SOR			
	w	k	t	e	w	k	t	e	w	k	t	e
45	1.683	55	0.004	5.41E-06	1.633	38	0.004	3.91E-06	1.622	32	0.000	3.85E-06
85	1.764	98	0.014	5.66E-06	1.702	58	0.016	3.98E-06	1.604	44	0.005	3.87E-06
105	1.789	124	0.024	5.84E-06	1.758	73	0.033	4.31E-06	1.705	56	0.013	4.44E-06
145	1.851	163	0.036	5.91E-06	1.781	98	0.038	4.24E-06	1.724	84	0.027	4.62E-06
185	1.899	191	0.067	5.93E-06	1.784	116	0.069	3.66E-06	1.771	95	0.042	4.04E-06
225	1.914	265	0.134	5.68E-06	1.789	138	0.141	3.57E-06	1.783	123	0.097	3.99E-06

t is the execution time of the computer with corresponding *w* in seconds(s).

Table 1 shows the comparison of the results for standard five-point, nine-point group SOR and preconditioned nine-point group SOR iterative methods. The results show the corresponding number of iterations (k), value of optimum w obtained, and the maximum error (e).

Fig.3 shows the comparison of the number of iterations between these three methods. The graph explained that the preconditioned nine-point group SOR method gives the minimum number of iterations and the difference became obvious when the value of N increased. Where else, Fig.4 compares the CPU time for standard five-point, nine-point group SOR and preconditioned nine-point group SOR iterative methods. From the graph obtained in Fig.4, the nine-point group method requires the minimum number of iterations, but it did not give the lowest CPU time. This is because this method has larger amount of computing effort compared to the preconditioned nine-point group SOR iterative method which requires the minimum CPU time amongst other methods.

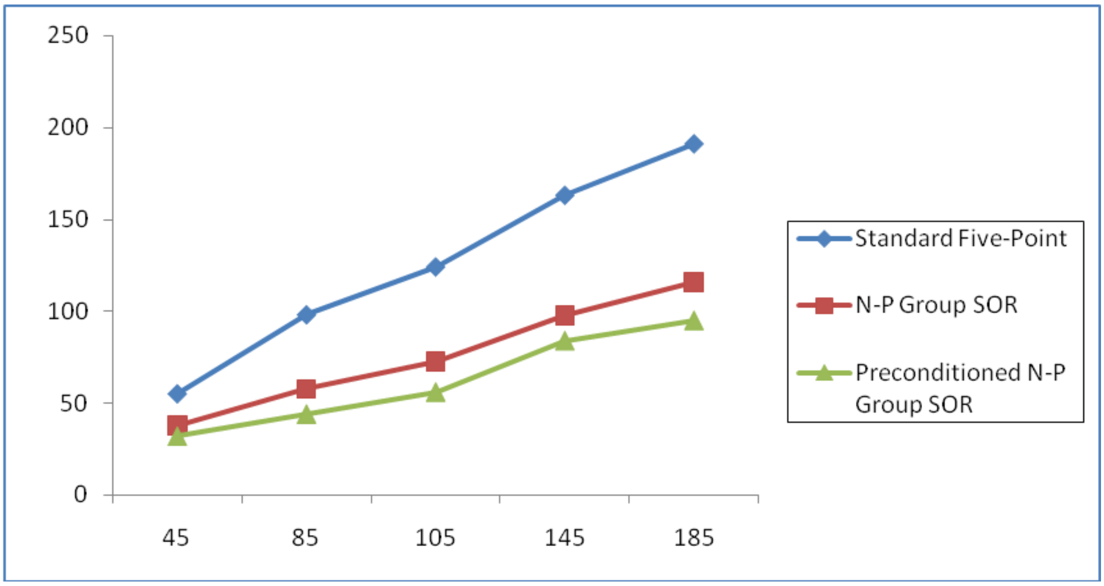

Fig. 3: Comparison of number of iterations (k) for Standard Five-Point, N-P SOR and preconditioned N-P SOR iterative methods

Since the convergence of the iteration methods relies on the spectral radius, which is defined as the largest of the moduli of the eigenvalues of the iteration matrix. It is stated and proven that a linear system with smaller value of spectral radius will have better convergence rate ([8], [9], [10]). Thus, the spectral radius of the coefficient matrix of the original system and the preconditioned system will be compared in order to justify the performance and suitability of the preconditioner. Since there are no special theoretical formulas that can be used to determine the spectral radiuses of the preconditioned matrices, therefore, we use Matlab software to estimate the values of the spectral radius.

Table 2 shows the comparison of the spectral radius between the original N-P Group SOR and the preconditioned N-P Group SOR systems. Clearly it can be seen that the spectral radius of the preconditioned system is smaller compared to the original system, thus justifying our findings.

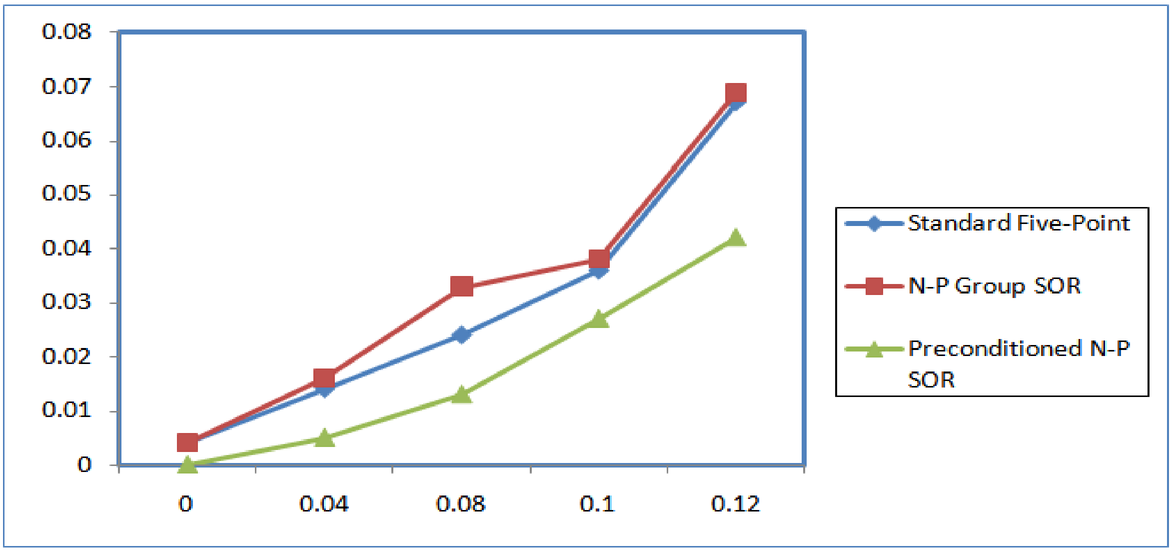

Fig. 4: Comparison of the CPU time (t) for Standard Five-Point, N-P SOR and preconditioned N-P SOR iterative methods

Table 2: Comparison of spectral radius between the original and the preconditioned linear systems

N	Original N-P Group SOR system	Preconditioned N-P Group SOR system
45	0.7741	0.4231
85	0.8322	0.4603
105	0.8603	0.4934
145	0.8943	0.5022
185	0.9402	0.5791
225	0.9474	0.6343

V. CONCLUSION

The application of the new preconditioner in block formulation for the N-P Group SOR iterative method is presented to accelerate the convergence rate of this group method. We see that the resulted preconditioned system showed improvements in the number of iterations and the execution time. Hence, we conclude that the proposed preconditioner is suitable to be implemented on the N-P Group SOR method and is able to accelerate the rate of convergence of this method. For future work, it would be worthwhile effort to investigate the application of preconditioner for other types of explicit group methods.

ACKNOWLEDGEMENTS

Financial support provided by Qassim University for the completion of this research is gratefully acknowledged.

REFERENCES

[1] Y. Saad and H. A. Van der Vorst, Iterative Solution of Linear Systems in the 20th Century, *Journal of Computational and Applied Math, 123,* 2000, 1-33.

[2] W. S. Yousif and D. J. Evans, Explicit Group Over Relaxation Methods for Solving Elliptic Partial Differential Equations, *Mathematics and Computers in Simulation, 28,* 1986, 453-466.

[3] M. M. Martins, D. J. Evans and W. Yousif, Further results on the preconditioned SOR method, *International Journal of Computer Mathematics, 77(4),* 2001, 603-610.

[4] N. H. M. Ali and A. M. Saeed, Convergence Analysis of the Preconditioned Group Splitting Methods in Boundary Value Problems, *Abstract and Applied Analysis, 2012,* 2012, 1-14.

[5] A. M. Saeed, Fast Iterative Solver for The 2-D Convection Diffusion Equations, *Journal Of Advances In Mathematics, 9 (6),* 2014, 2773-2782.

[6] A. M. Saeed, Efficient Group Iterative Method for Solving the Biharmonic Equation, *British Journal of Mathematics & Computer Science, 9 (3),* 2014, 237-245.

[7] S. C. Lee, *Point and group iterative method accelerated techniques for solving the Poisson problem*, Master's thesis, USM, Malaysia, 2006.

[8] A. M. Saeed and N. H. M. Ali, Preconditioned $(I+\bar{S})$ Group Iterative Methods On Rotated Grids, *European Journal of Scientific Research, 37(2),* 2009, 278-287.

[9] D.M. Young, *Iterative solutions of large linear systems* (New York, Academic Press, 1971).

[10] D.S. Watkins, *Fundamentals of Matrix Computations2nd edn.* (New York, John Wiley & Sons, 2002).