Explicit group USSOR method for solving elliptic partial differential equations

EXPLICIT GROUP USSOR METHOD FOR SOLVING ELLIPTIC PARTIAL DIFFERENTIAL EQUATIONS

W. S. YOUSIF[1], J. L. SANTOS[2,3] and M. M. MARTINS[4,2]

[1]Department of Computer Science, Loughborough University, Loughborough, LE11 3TU, England, UK.
[2]Department of Mathematics, University of Coimbra, 3000 Coimbra, Portugal.
[3]CMUC, Department of Mathematics, University of Coimbra, 3001-454 Coimbra, Portugal.
[4]Institute of Telecommunications, University of Coimbra, 3000 Coimbra, Portugal

ABSTRACT. This paper presents a new 4-points Explicit Group Unsymmetric Successive Overrelaxation (USSOR) iterative method to approximate the solution of the linear systems derived from the discretisation of self-adjoint elliptic partial equations. Several studies have been carried out by many researchers on the USSOR iterative method, for example, the analysis of its convergence [1], an upper bound for its error [2] and recently a special case of the USSOR, namely the SSOR method has been used to approximate the solution of augmented systems [4] and [8]. The computational behaviour of this new method and a comparison with its point version is presented.

Key Words: USSOR method; elliptic partial differential equation; group iterative methods, five-point approximation scheme, Laplace's equation.

AMS (MOS) Subject Classification. 65F10

1. INTRODUCTION AND PRELIMINARIES

Consider the linear system of equations

$$A\,x = b, \tag{1}$$

where $A \in C^{n,n}$ is a given non-singular matrix with non vanishing diagonal entries, $b \in C^{n}$ is a known vector and x is an unknown vector.

Many iterative methods are normally used to obtain an approximation for the solution of (1), one of these iterative methods is the Unsymmetric Successive Overrelaxation (USSOR) method [5, 6], which will be defined in the following. This iterative method can be used if the block diagonal part of the coefficient matrix A of the system (1) is non singular. Some authors have enlarged the convergence region of the USSOR method [1], others [2], have obtained an upper bound of its error and recently a variant of it, i.e., the SSOR iterative method has been used to approximate the solution of augmented systems, namely, the solution of Navier–Stokes problem, [4, 8]. In this paper we will use the 4-points explicit group USSOR iterative method to approximate the solution of the linear self- adjoint elliptic equation,

$$\frac{\partial}{\partial x}\left(A(x,y)\frac{\partial U}{\partial x}\right)+\frac{\partial}{\partial y}\left(B(x,y)\frac{\partial U}{\partial y}\right)-F(x,y)U=G(x,y),\qquad (x,y)\in\Omega \tag{2}$$

$$U(x,y) \;=\; g(x,y),\qquad (x,y)\in\partial\Omega \tag{3}$$

defined in a bounded region Ω, where $A(x, y) > 0$, $B(x, y) > 0$ and $F(x,y) \geq 0$ and $\partial\Omega$ is the boundary of Ω.

The discretisation of (2) leads to (1), [5, 6]. Therefore, let us consider

$$A = D - E - F \tag{4}$$

where $D = \mathrm{diag}(A)$, E and F are strictly lower and upper triangular matrices obtained from A, respectively.

The USSOR iterative method is given by:

$$(D-\;\omega E)\,u^{(k+\frac{1}{2})}=\left[(1-\omega)D+\omega F\right]u^{(k)}+\omega\, b,\qquad k=0,1,\ldots \tag{5}$$

and

$$(D-\;\omega' F)\,u^{(k+1)}=\left[(1-\omega')D+\omega' E\right]u^{(k+\frac{1}{2})}+\omega'\, b,\qquad k=0,1,\ldots \tag{6}$$

where ω and ω' are real non-null parameters.

If we define $L = D^{-1}E$ and $U = D^{-1}F$ then the equations (5) and (6) can be written as

$$u^{(k+1)}=S_{\omega,\omega'}u^{(k)}+(\omega+\omega'-\omega\omega')(I-\omega' U)^{-1}(I-\omega L)^{-1}\,b,\qquad k=0,1,\ldots \tag{7}$$

where $\quad S_{\omega,\omega'} = U_{\omega'}L_{\omega}$ (8)

with

$$L_{\omega} = (I - \omega L)^{-1}\left[(1-\omega)I + \omega U\ \right],\tag{9}$$

and

$$U_{\omega'} = (I - \omega' U)^{-1}\left[(1-\omega')I + \omega' L\ \right].\tag{10}$$

If $\omega = \omega'$ in (7), then the Symmetric Successive Overrelaxation (SSOR) iterative method is obtained.

In the following, and for simplicity, we will consider Laplace's equation

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = 0 \tag{11}$$

defined in the unit square, $0 \le x, y \le 1$, with m^2 internal mesh points. It can be easily seen that equation (11) is a special case of equation (2) if we consider $A(x, y) = B(x, y) = 1$ and $F(x, y) = G(x, y) = 0$.

The standard technique for solving the sparse linear systems derived from the discretisation of self-adjoint elliptic partial differential equations by finite difference techniques (block or line iterative methods) can be improved if we use explicit group iterative methods [7].

Therefore, in this paper, new explicit 4-points group USSOR iterative method is presented. A comparison between the point USSOR method [5, 6], and the 4-points group USSOR method for the solution of the model problem is made and the behaviour of this new method is discussed.

In the sequel let us consider the linear system (1) with the matrix A having Property $\mathrm{A}^{(\pi)}$ and being π-consistently ordered. Therefore, we will present some definitions, given in [6].

Definition 1.1

An ordered grouping π of $W = \{1, 2,\dots, n\}$ is a subdivision of W into disjoint subsets R_1, R_2, $\dots$, R_q such that $R_1 + R_2 + \dots + R_q = W$.

Given a matrix A and an ordered grouping π we define the sub matrices $A_{m,n}$ for m, $n = 1, 2,\dots, q$ as follows: $A_{m,n}$ is formed from A deleting all rows except those corresponding to R_m and all columns except those corresponding to R_n.

Definition 1.2

Let π be an ordered grouping with q groups. A matrix A has Property $\mathrm{A}^{(\pi)}$ if the $q \times q$ matrix $Z = (z_{r,s})$ defined by

$$z_{r,s} = \begin{cases} 0 & if\ \ A_{r,s} = 0 \\ 1 & if\ \ A_{r,s} \neq 0 \end{cases} \tag{12}$$

has Property A.

Definition 1.3

A matrix A of order n is consistently ordered if for some t there exist disjoint subsets S_1, S_2, …, S_t of $W=\{1, 2,\ldots, n\}$ such that $\sum_{k=1}^{t} S_k = W$ and such that if i and j are associated, then $j \in S_{k+1}$ if $j > i$ and $j \in S_{k-1}$ if $j < i$, where S_k is the subset containing i.

Definition 1.4

A matrix A is a π- consistently ordered matrix if the matrix Z is consistently ordered.

2. THE 4-POINTS GROUP USSOR ITERATIVE METHOD

In this section we will present an explicit set of equations for the 4-points group USSOR iterative method [7], where each group is formed from 4 points of the net region, according to Figure 1, where $t = (qm+1)$, step 2, $(q+1)m-1$, m is an even number and $q = 0$, step 2, m-2. Each group G_k, $k = 1, 2, \ldots, m^2/4$ contains only four elements $\{t, t+1, t+m, t+m+1\}$ ordered column wise.

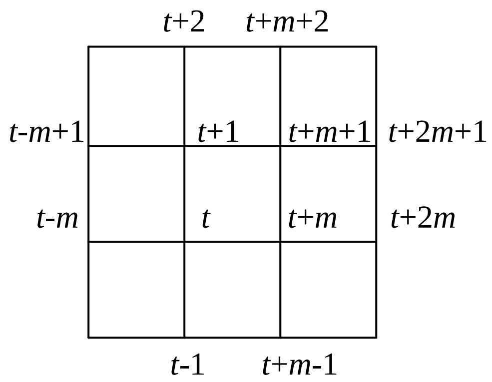


Figure 1

Suppose that the groups are ordered in red-black ordering (see Figure 2) in the case where the mesh is the unit square and $\Delta x = \Delta y = h = 1/5$.

If the five-point approximation scheme is used then the finite difference equation at the point P (see Figure 3) has the form

$$u_p + \alpha_1 u_{B,P} + \alpha_2 u_{R,P} + \alpha_3 u_{T,P} + \alpha_4 u_{L,P} = b_P, \tag{13}$$

where B, R, T and L denote Bottom, Right, Top and Left of the point P, respectively.

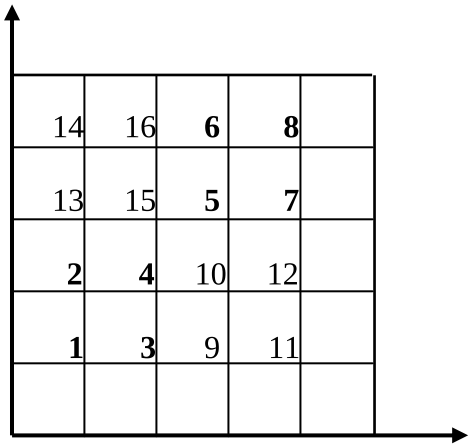


Figure 2

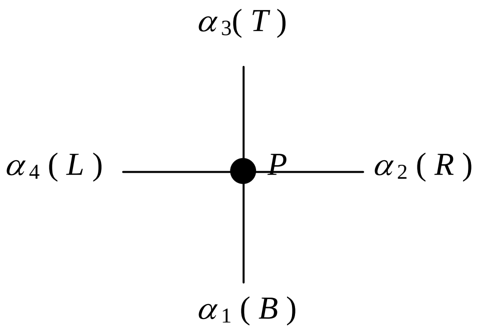


Figure 3

If this scheme is used, for all the mesh points, we have the linear system

$A_1 u = b_1$

with

$$A_1 = \begin{pmatrix} R_0 & 0 & R_2 & R_3 \\ 0 & R_0 & R_1 & R_4 \\ R_4 & R_3 & R_0 & 0 \\ R_1 & R_2 & 0 & R_0 \end{pmatrix} \tag{14}$$

where $R_0 = \left(\begin{array}{cc:cc} 1 & \alpha_3 & \alpha_2 & 0 \\ \alpha_1 & 1 & 0 & \alpha_2 \\ \hdashline \alpha_4 & 0 & 1 & \alpha_3 \\ 0 & \alpha_4 & \alpha_1 & 1 \end{array}\right)$, $R_1 = \left(\begin{array}{cc:cc} 0 & \alpha_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hdashline 0 & 0 & 0 & \alpha_1 \\ 0 & 0 & 0 & 0 \end{array}\right)$, $R_2 = \left(\begin{array}{cc:cc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hdashline \alpha_2 & 0 & 0 & 0 \\ 0 & \alpha_2 & 0 & 0 \end{array}\right)$,

$$R_3 = \left(\begin{array}{cc:cc} 0 & 0 & 0 & 0 \\ \alpha_3 & 0 & 0 & 0 \\ \hdashline 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha_3 & 0 \end{array}\right) \text{ and } R_4 = \left(\begin{array}{cc:cc} 0 & 0 & \alpha_4 & 0 \\ 0 & 0 & 0 & \alpha_4 \\ \hdashline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right). \tag{15}$$

The matrix A_1 (in (14)), has Property $A^{(\pi)}$ and is π- consistently ordered.

To derive the explicit 4-points group USSOR method, we evaluate the transformed matrix A_2 and the modified vector b_2, where

$$A_2 = T^{-1} A_1 \tag{16}$$

and

$$b_2 = T^{-1} b, \tag{17}$$

where $T = \mathrm{diag}\{R_0\}$.

As T^{-1} is equal to diag{ R_0^{-1} } and the matrix R_0^{-1} is given by

$$R_0^{-1} = \frac{1}{(\alpha_6+1)^2+2\alpha_5-1}\begin{bmatrix} \alpha_5 & \alpha_3\alpha_6 & \alpha_2\alpha_7 & 2\alpha_2\alpha_3 \\ \alpha_1\alpha_6 & \alpha_5 & 2\alpha_1\alpha_2 & \alpha_2\alpha_7 \\ \alpha_4\alpha_7 & 2\alpha_3\alpha_4 & \alpha_5 & \alpha_3\alpha_6 \\ 2\alpha_1\alpha_4 & \alpha_4\alpha_7 & \alpha_1\alpha_6 & \alpha_5 \end{bmatrix}, \tag{18}$$

where

$$\alpha_5 = 1-\alpha_1\alpha_3-\alpha_2\alpha_4,\ \ \alpha_6 = \alpha_1\alpha_3-\alpha_2\alpha_4-1,\ \ \alpha_7 = \alpha_2\alpha_4-\alpha_1\alpha_3-1.$$

Therefore,

$$A_2 = \begin{bmatrix} I & C \\ B & I \end{bmatrix} \tag{19}$$

where C and B can be evaluated easily.

The matrices A_1 and A_2 have the same block structures. The unique difference is that instead of the matrices R_0 and R_i, i = 1, …, 4 we have the identity matrices and $R_0^{-1} R_i$, respectively.

For the model problem and a square grid, we have

$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = -\frac{1}{4}.$$

Therefore, from (18)

$$R_0^{-1} = \frac{1}{6}\begin{bmatrix} 7 & 2 & 2 & 1 \\ 2 & 7 & 1 & 2 \\ 2 & 1 & 7 & 2 \\ 1 & 2 & 2 & 7 \end{bmatrix}, \tag{20}$$

and hence,

$$R_0^{-1}R_1 = -\frac{1}{24}\begin{bmatrix} 0 & 7 & 0 & 2 \\ 0 & 2 & 0 & 1 \\ 0 & 2 & 0 & 7 \\ 0 & 1 & 0 & 2 \end{bmatrix}. \tag{21}$$

$R_0^{-1} R_i$, i = 2, 3, 4 can be obtained in a similar way. Thus, the computational molecule at the point P can be set up, as it is shown in Figure 4.

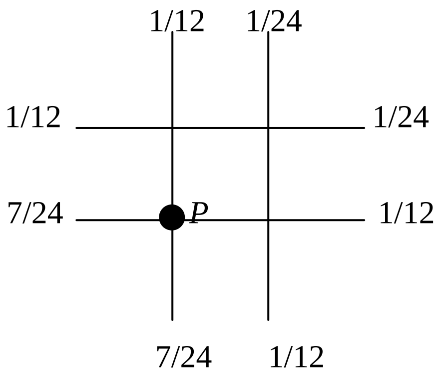


Figure 4

Therefore we can derive the explicit 4- points group USSOR iterative method, by using this molecule:

$$y_t^{(k+1)} = \frac{1}{24}\Big\{7\Big[\omega(y_{t-m}^{(k+1)} + y_{t-1}^{(k+1)})\Big] + 2\Big[\omega(u_{t+2m}^{(k)} + u_{t+2}^{(k)} + y_{t+m-1}^{(k+1)} + y_{t-m+1}^{(k+1)})\Big] + \\ + \omega(u_{t+m+2}^{(k)} + u_{t+2m+1}^{(k)})\Big\} + (1-\omega)u_t^{(k)} \quad (22a)$$

$$y_{t+1}^{(k+1)} = \frac{1}{24}\Big\{7\Big[\omega(y_{t-m+1}^{(k+1)} + u_{t+2}^{(k)})\Big] + 2\Big[\omega(y_{t-m}^{(k+1)} + y_{t-1}^{(k+1)} + u_{t+2m+1}^{(k)} + u_{t+m+2}^{(k)})\Big] + \\ + \omega(y_{t+m-1}^{(k+1)} + u_{t+2m}^{(k)})\ \Big\} + (1-\omega)u_{t+1}^{(k)} \quad (22b)$$

$$y_{t+m}^{(k+1)} = \frac{1}{24}\Big\{7\Big[\omega(y_{t+m-1}^{(k+1)} + u_{t+2m}^{(k)})\Big] + 2\Big[\omega(y_{t-m}^{(k+1)} + y_{t-1}^{(k+1)} + u_{t+2m+1}^{(k)} + u_{t+m+2}^{(k)})\Big] + \\ + \omega(y_{t-m+1}^{(k+1)} + u_{t+2}^{(k)})\ \Big\} + (1-\omega)u_{t+m}^{(k)} \quad (22c)$$

$$y_{t+m+1}^{(k+1)} = \frac{1}{24}\Big\{7\Big[\omega(u_{t+2m+1}^{(k)} + u_{t+m+2}^{(k)})\Big] + 2\Big[\omega(y_{t+m-1}^{(k?+1)} + y_{t-m+1}^{(k+1)} + u_{t+2m}^{(k)} + u_{t+2}^{(k)})\Big] + \\ + \omega(y_{t-m}^{(k+1)} + y_{t-1}^{(k+1)})\ \Big\} + (1-\omega)u_{t+m+1}^{(k)} \quad (22d)$$

where $t = (pm + 1)$, step 2, $(p + 1)m - 1$ and $p = 0$, step 2, m-2, and

$$u_t^{(k+1)} = \frac{1}{24}\Big\{7\Big[\omega'(y_{t-m}^{(k+1)} + y_{t-1}^{(k+1)})\Big] + 2\Big[\omega'(u_{t+2m}^{(k+1)} + u_{t+2}^{(k+1)} + y_{t+m-1}^{(k+1)} + y_{t-m+1}^{(k+1)})\Big] + \\ + \omega'(u_{t+m+2}^{(k+1)} + u_{t+2m+1}^{(k+1)})\Big\} + (1-\omega')y_t^{(k+1)} \quad (23a)$$

$$u_{t+1}^{(k+1)} = \frac{1}{24}\Big\{7\Big[\omega'(y_{t-m+1}^{(k+1)} + u_{t+2}^{(k+1)})\Big] + 2\Big[\omega'(y_{t-m}^{(k+1)} + y_{t-1}^{(k+1)} + u_{t+2m+1}^{(k+1)} + u_{t+m+2}^{(k+1)})\Big] + \\ + \omega'(y_{t+m-1}^{(k+1)} + u_{t+2m}^{(k+1)})\ \Big\} + (1-\omega')y_{t+1}^{(k+1)} \quad (23b)$$

$$u_{t+m}^{(k+1)} = \frac{1}{24}\Big\{7\Big[\omega'(y_{t+m-1}^{(k+1)} + u_{t+2m}^{(k+1)})\Big] + 2\Big[\omega'(y_{t-m}^{(k+1)} + y_{t-1}^{(k+1)} + u_{t+2m+1}^{(k+1)} + u_{t+m+2}^{(k+1)})\Big] + \\ + \omega'(y_{t-m+1}^{(k+1)} + u_{t+2}^{(k+1)})\ \Big\} + (1-\omega')y_{t+m}^{(k+1)} \quad (23c)$$

$$u_{t+m+1}^{(k+1)} = \frac{1}{24}\Big\{7\Big[\omega'(u_{t+2m+1}^{(k+1)} + u_{t+m+2}^{(k+1)})\Big] + 2\Big[\omega'(y_{t+m-1}^{(k?+1)} + y_{t-m+1}^{(k+1)} + u_{t+2m}^{(k+1)} + u_{t+2}^{(k+1)})\Big] + \\ + \omega'(y_{t-m}^{(k+1)} + y_{t-1}^{(k+1)})\ \Big\} + (1-\omega')y_{t+m+1}^{(k+1)} \quad (23d)$$

where t =(p + 1)m – 1, step -2, (pm + 1), and p = m-2, step -2, 0.

If $\omega = \omega'$ we have the explicit group SSOR iterative method.

3. ANALYSIS OF THE POINT USSOR AND 4-POINTS GROUP USSOR METHODS

The relative efficiency of the point USSOR method and the 4-points group USSOR method will be discussed in this section.
In both methods we assume that there are m^2 internal mesh points in the solution domain. We also assume that the execution times for the addition and multiplication operations are roughly the same. The number of iterations and CPU time needed to approximate the solution of (1) will be obtained computationally in Section 4, using the formulas presented in Sections 3.1 and 3.2 for the point USSOR and the 4-points group USSOR methods, respectively.

3.1 The Point USSOR Method

The finite difference solution of the model problem by the point USSOR iterative method is given by

$$y_t^{(k+1)} = \omega_1 (y_{t-1}^{(k+1)} + y_{t-m}^{(k+1)} + u_{t+1}^{(k)} + u_{t+m}^{(k)}) + \omega_2 u_t^{(k)}, \qquad t = 1,\ldots,m^2$$

and (24)

$$u_t^{(k+1)} = \omega'_1 (y_{t-1}^{(k+1)} + y_{t-m}^{(k+1)} + u_{t+1}^{(k+1)} + u_{t+m}^{(k+1)}) + \omega'_2\, y_t^{(k+1)}, \qquad t = m^2,\ldots,1$$

where $\omega_1 = \dfrac{\omega}{4}, \omega'_1 = \dfrac{\omega'}{4}, \quad \omega_2 = 1-\omega$ and $\omega'_2 = 1-\omega'$.

By assuming that $\omega_1, \omega'_1, \omega_2$ and ω'_2 are stored beforehand, it can be observed that the number of operations required (excluding the convergence test) for the point USSOR iterative method is 12 m^2 operations per iteration. With the equalities (24) we can easily obtain computational results for this method.

3.2 The 4-Points Group USSOR Method

To calculate the number of operations and CPU time, per iteration, using the 4-points group USSOR method to approximate the solution of the model problem (1), it can be seen, from equations (22) and (23), that the required number of operations (excluding the convergence test) is 24 m^2 operations per iteration. However, it can be noticed, from equations (22) and (23), that not all the elements involved in the calculations of the four points are different, then the number of operations can be reduced to 17 m^2 operations per iteration as shown bellow.

In the forward step, i.e. equation (22), if we set

$$s_1 = u_{t+2m}^{(k)} + u_{t+2}^{(k)}, \quad s_2 = u_{t+2m+1}^{(k)} + u_{t+m+2}^{(k)}, \quad s_3 = y_{t-m}^{(k+1)} + y_{t-1}^{(k+1)}, \; s_4 = y_{t+m-1}^{(k+1)} + y_{t-m+1}^{(k+1)},$$

$$s_5 = y_{t-m+1}^{(k+1)} + u_{t+2}^{(k)}, \quad s_6 = y_{t+m-1}^{(k+1)} + u_{t+2m}^{(k)}, \quad t_1 = 2(s_1 + s_4), \quad t_2 = 2(s_2 + s_3)$$

then we have

$$y_t^{(k+1)} = \omega_1(7s_3 + t_1 + s_2) + \omega_2 u_t^{(k)}$$
$$y_{t+1}^{(k+1)} = \omega_1(7s_5 + t_2 + s_6) + \omega_2 u_{t+1}^{(k)} \quad , \quad t = (pm+1), \text{ step } 2, (p+1)m - 1 \text{ and}$$
$$y_{t+m}^{(k+1)} = \omega_1(7s_6 + t_2 + s_5) + \omega_2 u_{t+m}^{(k)} \qquad p = 0, \text{ step } 2, m\text{-}2 \tag{25}$$
$$y_{t+m+1}^{(k+1)} = \omega_1(7s_2 + t_1 + s_3) + \omega_2 u_{t+m+1}^{(k)}$$

Similarly, for the backward step, equation (23), we set

$$s_1 = u_{t+2m}^{(k+1)} + u_{t+2}^{(k+1)}, \quad s_2 = u_{t+2m+1}^{(k+1)} + u_{t+m+2}^{(k+1)}, \quad s_3 = y_{t-m}^{(k+1)} + y_{t-1}^{(k+1)}, \quad s_4 = y_{t+m-1}^{(k+1)} + y_{t-m+1}^{(k+1)},$$
$$s_5 = y_{t-m+1}^{(k+1)} + u_{t+2}^{(k+1)}, \quad s_6 = y_{t+m-1}^{(k+1)} + u_{t+2m}^{(k+1)}, \quad t_1 = 2(s_1 + s_4), \quad t_2 = 2(s_2 + s_3),$$

then we have

$$u_t^{(k+1)} = \omega'_1(7s_3 + t_1 + s_2) + \omega'_2\, y_t^{(k+1)}$$
$$u_{t+1}^{(k+1)} = \omega'_1(7s_5 + t_2 + s_6) + \omega'_2\, y_{t+1}^{(k+1)} \quad , t = (p+1)m\text{-}1, \text{ step -}2, (pm+1) \quad \text{and}$$
$$u_{t+m}^{(k+1)} = \omega'_1(7s_6 + t_2 + s_5) + \omega'_2\, y_{t+m}^{(k+1)}, \qquad p = m\text{-}2, \text{ step -}2, 0 \tag{26}$$
$$u_{t+m+1}^{(k+1)} = \omega'_1(7s_2 + t_1 + s_3) + \omega'_2\, y_{t+m+1}^{(k+1)},$$

where $\omega_1 = \dfrac{\omega}{24}, \omega'_1 = \dfrac{\omega'}{24}$, $\omega_2 = 1 - \omega$ and $\omega'_2 = 1 - \omega'$. These parameters need only be calculated once.

As mentioned before, if $\omega = \omega'$, then from equations (24) and (25), the 4-points group SSOR iterative method is obtained.

4. NUMERICAL RESULTS

In order to compare the point USSOR and 4-points group USSOR iterative methods we will present, in this section, some numerical results.

The numerical experiments have been performed using Matlab 7.9, on Core 2 Duo, 2.26 GHZ (4GM RAM), laptop (MacBook Pro) with Macintosh system. The methods have been compared in terms of number of iterations and CPU time (in seconds). Throughout the experiments the convergence test used was the average error test with tolerance error $\varepsilon = 10^{-7}$.

Problem 1. Firstly, the two methods were applied to approximate the solution of Laplace's equation

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = 0 \qquad (x,y) \in \Omega = (0,1) \times (0,1) \tag{27}$$

with the Dirichlet boundary conditions

$$\begin{aligned} &U(x,0) = \sin \pi x, && 0 \le x \le 1, \\ &U(0,y) = U(1,y) = U(x,1) = 0, && 0 \le x, y \le 1 \end{aligned} \tag{28}$$

The numerical solution of the problem (27)-(28), using the 4-points group USSOR iterative method, whit $h = 1/13$, is illustrated in Figure 5.

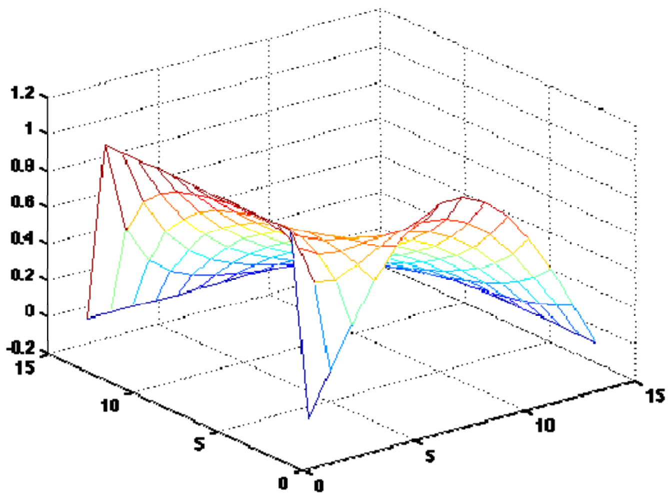


Figure 5: Numerical solution of the problem (27)-(28) obtained with h = 1/13.

The experimental optimum values of ω and ω' were determined to within ±0.01 by solving the problem for a range of values of ω and ω' then choosing those which gives the minimum number of iterations. The obtained results, i.e., the experimental optimum values of ω and ω', the minimum number of iterations and the CPU time in seconds required to solve problem (27)-(28) are summarized in Table 1 and Table 2 for the point USSOR and the 4-points group USSOR methods, respectively.

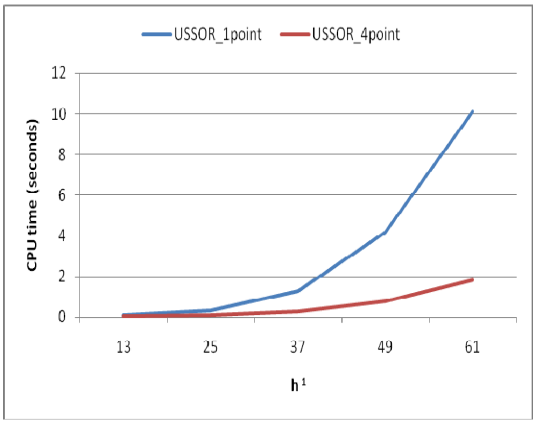
Table 1**:** Computational results of the point USSOR method for the problem (27)-(28)

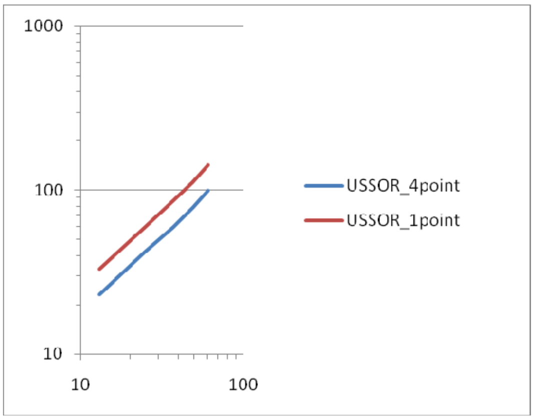
h^{-1}	ω	ω'	No. of Iterations	CPU time (seconds)
13	0.88-0.91	1.65	33	0.08
25	1.13-1.15	1.80	60	0.34
37	1.28-1.29	1.86	86	1.26
49	1.64	1.90	112	4.16
61	1.83	1.93	143	10.12

Table 2**:** Computational results of the 4-points group USSOR method for the problem (27)-(28)

h^{-1}	ω	ω'	No. of Iterations	CPU time (seconds)
13	0.12	1.53	23	0.07
25	0.28	1.77	42	0.08
37	0.28	1.86	60	0.29
49	0.31	1.92	78	0.78
61	0.39	1.98	100	1.81

The plots of the CPU computation time vs the mesh size for the two methods are given in Figure 6(a). Also, for the two methods, the logarithm of the number of iterations is plotted against $\log h^{-1}$, the graphs are shown in Figure 6(b). As expected, the plots for the two methods are straight lines with a slope of unity, thus verifying the SOR theory.





(a) (b)

Figure 6: Computational results for the point USSOR and the 4-point group USSOR methods with the computational optimal parameters presented in Tables 1 and 2.

From these results we can conclude that the 4-point group USSOR method saves 70% - 80% of the CPU time in comparison with the corresponding version of the point USSOR method. We can also notice that the number of iterations increases linearly with the problem size while the increase in the CPU time is quadratic.

Problem 2. Numerical experiments were also carried out on solving Laplace's equation (27) using the Dirichlet boundary conditions

$$\begin{aligned} &U(0, y) = 100, && 0 \le y \le 1, \\ &U(x, 0) = U(x, 1) = U(1, y) = 0, && 0 \le x, y \le 1. \end{aligned} \tag{29}$$

The numerical solution for the problem (27)-(29), using the 4-points group USSOR iterative method with $h = 1/13$, is illustrated in Figure 7.

The experimental optimum values of ω and ω', the minimum number of iterations and the CPU time in seconds were obtained in a similar manner to Problem 1 and the results are summarized in Table 3 and Table 4 for the point USSOR and the 4-points group USSOR methods, respectively.

The plots of the CPU computation time vs the mesh size for the two methods are given in Figure 8(a). Again, for the two methods, the logarithm of the number of iterations is plotted against $\log h^{-1}$, the graphs are shown in Figure 8(b). The plots for the two methods were also straight lines with a slope of unity, thus verifying the SOR theory.

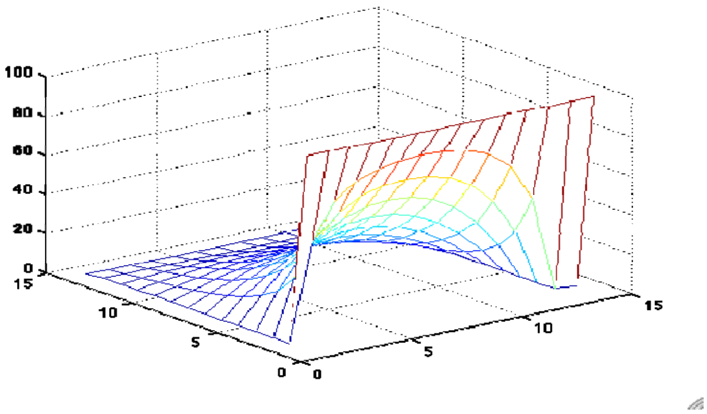


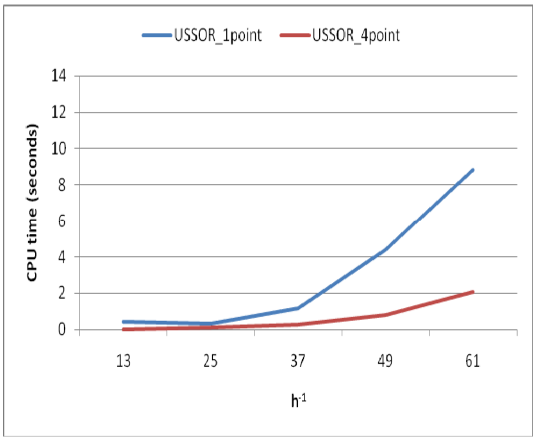
Figure 7: Numerical solution of the problem (27)-(29)

Table 3**:** Computational results of the point USSOR method for the problem (27)-(29)

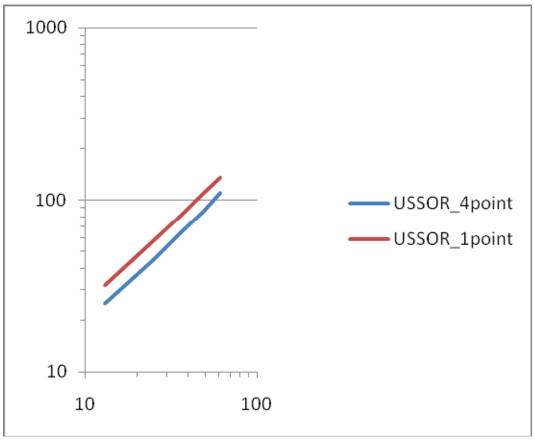
h^{-1}	ω	ω'	No. of Iterations	CPU time (seconds)
13	1.63	0.48	32	0.44
25	1.79	0.85	58	0.34
37	1.85	0.71-0.73	83	1.18
49	1.89	1.40	110	4.37
61	1.91	1.44	135	8.84

Table 4**:** Computational results of the 4-points group USSOR method for the problem (27)-(29).

h^{-1}	ω	ω'	No. of Iterations	CPU time (seconds)
13	1.69	0.69	25	0.02
25	1.86	0.55	45	0.09
37	1.85	0.25	66	0.29
49	1.90	0.25	86	0.80
61	1.90	0.14	110	2.09



(a)



(b)

Figure 8: Computational results for the point USSOR and the 4-points group USSOR methods with the computational optimal parameters presented in Tables 3 and 4.

Similar conclusions to those given for problem (27)-(28) can also be obtained for this problem. Additionally, we would like to point out that if we alternate the values of the optimal parameters ω and ω' then we obtain a slightly higher number of iterations.

5. CONCLUSIONS

The results, given in Tables 1, 2, 3 and 4, show that the new 4-point group USSOR method offer significant economies over the point USSOR method, in comparison a saving of 70% - 80% of the CPU time was achieved.

Further, the 4-points group USSOR method is an explicit method and is suitable for parallel computers as it possesses separate and independent tasks, as the groups of 4- points can be executed concurrently.

Other blocks (groups) can also be considered, i.e., the 2, 9, 16 or 25 points group, however this will be matter of further research.

REFERENCES

[1] M. M. Martins and L. Khrisna, Some New Results on the Convergence of the SSOR and USSOR Methods, *Linear Algebra and its Applications*, Vol 106, 185-193, 1988.

[2] M. M. Martins, M. E. Trigo and M. M. Santos, An Error Bound for the SSOR and USSOR methods, *Linear Algebra and its Applications*, Vol 232, 131-147, 1996.

[3] M. M. Martins, W. S. Yousif and D. J. Evans, Explicit Group AOR Method for solving Elliptic Partial Equations, *Neural, Parallel and Scientific Computations*, Vol 10, 411-422, 2002.

[4] M. M. Martins, W. S. Yousif and J. L. Santos, A Variant of the AOR Method for Augmented Systems, *Mathematics of Computation*, Vol 81, 399 -417, 2012.

[5] R. S. Varga, *Matrix iterative analysis*, Englewood Cliffs, NJ: Prentice-Hall, 1962.

[6] D. Young, *Iterative solutions of large linear systems*. New York: Academic Press, 1971.

[7] W. S. Yousif and D. J. Evans, Explicit group over-relaxation methods for solving elliptic partial differential equations, *Maths and Computers in Simulation*, Vol 28, 453-466, 1986.

[8] B. Zheng, K. Wang and Y. Wu, SSOR- like methods for saddle point problems, *Int J. of Comp. Maths*, Vol 86, 1405-1423, 2009.