AN OPTIMAL ORDERING STRATEGY OF THE POINTS ITERATIVE ALGORITHM FOR SOLVING 2D CONVECTION-DIFFUSION PROBLEM

[1]Jumat Sulaiman, [2]Mohamed Othman & [3]Mohammad Khatim Hasan

[1]School of Science and Technology, Universiti Malaysia Sabah, Locked Bag 2073, 88999 Kota Kinabalu, Sabah.
[2]Dept. of Communication Technology and Network, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia
[3]Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, 43600 UKM Bangi, Selangor.

ABSTRACT. *In this paper, there are three ordering strategies considered such as lexicography (NA), red-black (RB) and four colour (4C) to be applied onto the family Successive Over-Relaxation (SOR) iterative methods, which are denoted as FSSOR, HSSOR and QSSOR respectively. Based on these strategies, it has shown that the RB strategy is the optimal among all strategies for the quarter-sweep iterative method to solve two-dimensional convection-diffusion equations by using the five points finite difference approximation equation. To confirm our assertion, we include numerical results obtained in order to show the efficiency of the QSSOR method with the RB strategy compared to the FSSOR and HSSOR methods.*

KEYWORDS. Convection-Diffusion Problems, Quarter-Sweep Iterative Approach, Red-Black Ordering Strategy.

INTRODUCTION

In the era of computer technology, the numerical techniques such as the finite difference, finite element, finite volume and boundary element methods play an important role in simulating a wide variety of science and engineering problems. Those methods have been employed by many researchers to gain approximate solutions. Apart of those methods, the findings on various iterative methods such as the full-, half- and quarter-sweep iterations (Evans & Yousif, 1990; Abdullah, 1991; Jumat & Abdul Rahman, 1998; Othman & Abdullah, 2000) are definitely important in solving any system of linear equations.

In this paper, however, we investigate the optimal ordering strategy to be applied onto the full-, half- and quarter-sweep iterative methods by using the Crank-Nicolson (CN) finite difference scheme in solving the heat transfer problem, mainly on a two-dimensional unsteady convection-diffusion problem. This is because of the combination of iterative schemes and ordering strategies, which have been proven, can accelerate the convergence rate, see Parter (1988), Evans and Yousif (1990) and Zhang (1996). To begin the derivation, let us consider the two-dimensional unsteady convection-diffusion problem as given by (Harbans & Abdul Rahman, 1996)

$$\frac{\partial U}{\partial t} - v\left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2}\right) + w\left(\frac{\partial U}{\partial x} + \frac{\partial U}{\partial y}\right) = F(x, y, t), \qquad (x, y, t) \in [0,1]\times[0,1]\times[0,T] \tag{1}$$

subject to the initial condition

$$U(x, y, 0) = g_1(x, y), \qquad (x, y) \in [0,1]\times[0,1]$$

and the Dirichlet boundary conditions

$$\left.\begin{array}{l} U(x,0,t)=g_2(x,t) \\ U(x,1,t)=g_3(x,t) \end{array}\right\}, \quad 0 \le x \le 1, \quad t>0$$

$$\left.\begin{array}{l} U(0,y,t)=g_4(y,t) \\ U(1,y,t)=g_5(y,t) \end{array}\right\}, \quad 0 \le y \le 1, \quad t>0$$

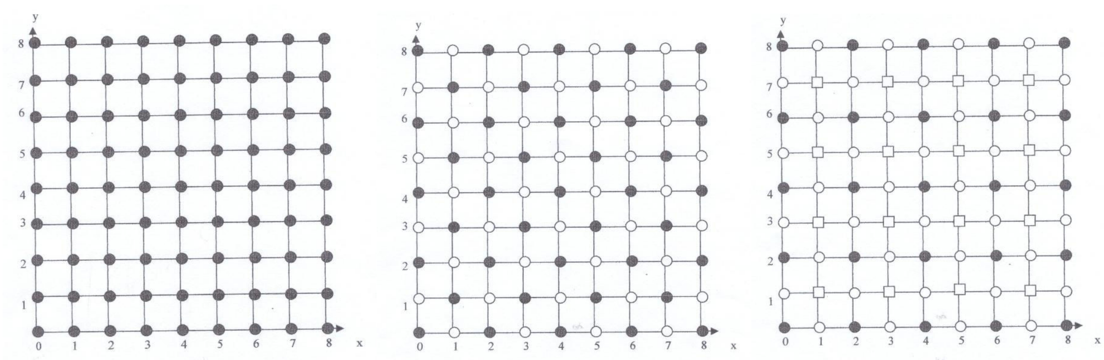
where v and w are diffusion and convection parameters, respectively. Then $U(x,y,t)$ is a function depend on the independent variables, x, y and t.

Before explaining on formulation of the finite difference approximation equation, in this paper we just consider in case of uniformly subinterval distances for each node point in the x and y directions, so that we can easily derive the full-, half- and quarter-sweep approximation equations for the problem defined in (1). Suppose that solution domain defined in (1) can be partitioned into $(m+1)$ subinterval in the x and y directions and $(n+1)$ in the t direction. The subintervals in the x, y, and t directions are denoted as Δx, Δy and Δt respectively which are defined as

$$\left.\begin{array}{l} \Delta x = \Delta y = h = \dfrac{1}{(m+1)} \\ \Delta t = \dfrac{T}{(n+1)} \end{array}\right\} \qquad (2)$$

FINITE GRID NETWORKS AND FINITE DIFFERENCE APPROXIMATION

In formulating various iterative schemes such as the full-, half- or quarter-sweep, we need to build the finite grid networks as a guide for development and implementation of the full-, half- and quarter-sweep algorithms mainly in applying several ordering strategies. Therefore, Figure 1 acts as a guide for development of the iterative schemes.



(a). (b). (c).

Figure 1. Finite grid networks for the (a) full-sweep, (b) half-sweep and (c) quarter-sweep in case of $m= 7$ at any level $t>0$.

Further more in this section, the Crank-Nicolson (CN) method is used to obtain the full-, half- and quarter-sweep approximation equations for the problem defined in (1). Suppose that the values of $U(x,y,t)$ at the point (x_i, y_j, t_k) can be approximated and indicated by $U_{i,j,k}$. By using the central difference, the full- and quarter-sweep approximation equations can be generally stated as

$$-\beta_2 U_{i-p,j,k+1} - \beta_2 U_{i,j-p,k+1} - \beta_3 U_{i,j+p,k+1} - \beta_3 U_{i+p,j,k+1} + \beta_1 U_{i,j,k+1} = f^{\beta}_{i,j,k} \qquad (3)$$

where,

$$\beta_1 = \left(\frac{1}{\Delta t} + \frac{4v}{2(ph)^2} \right), \quad \beta_2 = \left(\frac{v}{2(ph)^2} + \frac{w}{4(ph)} \right),$$

$$\beta_3 = \left(\frac{v}{2(ph)^2} - \frac{w}{4(ph)} \right), \quad \beta_4 = \left(\frac{1}{\Delta t} - \frac{4v}{2(ph)^2} \right),$$

$$f_{i,j,k}^{\beta} = \beta_2 (U_{i-p,j,k} + U_{i,j-p,k}) + \beta_3 (U_{i,j+p,k} + U_{i+p,j,k}) + \beta_4 U_{i,j,k} + F_{i,j,k}.$$

The value of p which corresponds to 1 and 2 represent case of the full- and quarter-sweep iteratives, respectively. Apart from equation (3), the rotated finite difference approximation equation (Dahlquist & Bjork, 1974) can be formed by the following transformation:

$$i, j \pm 1 \rightarrow i \pm 1, j \pm 1$$

$$i \pm 1, j \rightarrow i \pm 1, j \mp 1$$

$$\Delta x, \Delta y \rightarrow \sqrt{(\Delta x)^2 + (\Delta x)^2} = \sqrt{2}h$$

Using the above transformation, the distance in the i and j which correspond to $\Delta x = h$ and $\Delta y = h$ respectively become $\sqrt{2}h$. Therefore, the scheme of the CN method using the rotated finite difference approximation can be expressed as

$$-\alpha_2 U_{i-1,j-1,k+1} - \alpha_2 U_{i-1,j+1,k+1} - \alpha_3 U_{i+1,j-1,k+1} - \alpha_3 U_{i+1,j+1,k+1} + \alpha_1 U_{i,j,k+1} = f_{i,j,k}^{\alpha} \qquad (4)$$

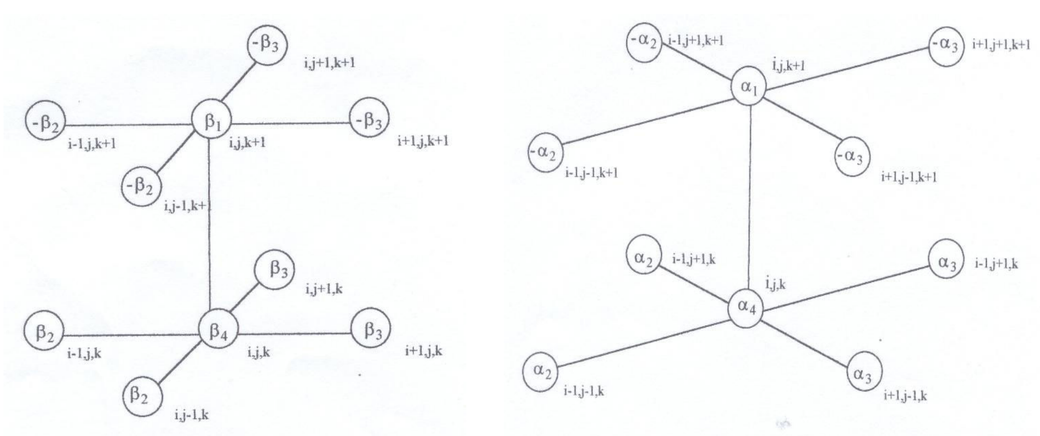
where,

$$\alpha_1 = \left(\frac{1}{\Delta t} + \frac{4v}{2(\sqrt{2}h)^2} \right), \quad \alpha_2 = \left(\frac{v}{2(\sqrt{2}h)^2} + \frac{w}{4\sqrt{2}h} \right),$$

$$\alpha_3 = \left(\frac{v}{2(\sqrt{2}h)^2} - \frac{w}{4\sqrt{2}h} \right), \quad \alpha_4 = \left(\frac{1}{\Delta t} - \frac{4v}{2(\sqrt{2}h)^2} \right),$$

$$f_{i,j,k}^{\alpha} = \alpha_2 (U_{i-1,j-1,k} + U_{i-1,j+1,k}) + \alpha_3 (U_{i+1,j-1,k} + U_{i+1,j+1,k}) + \alpha_4 U_{i,j,k} + F_{i,j,k}.$$

In fact the computational molecules for equations (3) and (4) based on the CN method can be shown in Figure 2.



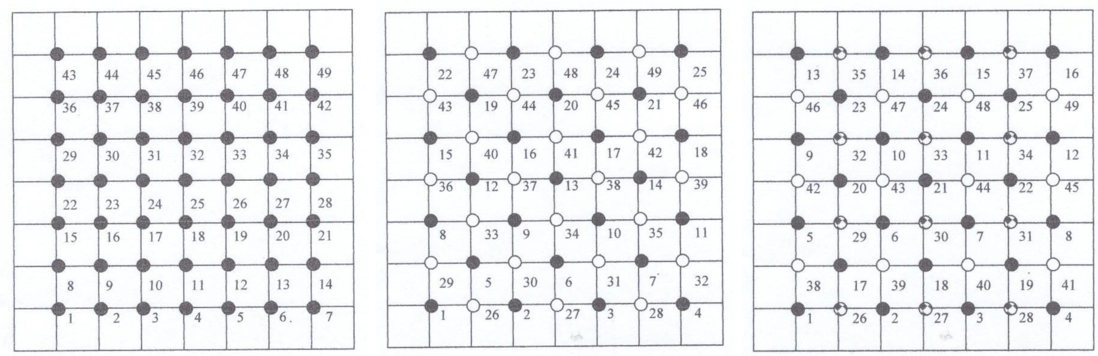
(a). (b).

Figure 2. Computational molecules of the CN method in case of (a) the full- and quarter-sweep and (b) the half-sweep for m = 7 at any level time $t > 0$.

IMPLEMENTATION OF ITERATIVE ORDERING STRATEGIES

In this section, there are three ordering strategies considered such as NA, RB and 4C to be applied onto the full-, half- and quarter-sweep SOR iterative methods, which are denoted as FSSOR, HSSOR and QSSOR respectively see Figure 3. The location of numbers 1, 2, 3, … , 49 for m = 7 shows on how the implementation of these methods will be computed by starting at number 1 and ending at number 49.

According to previous studies on the implementation of various orderings, it is obvious that combination of iterative schemes and ordering strategies which have been proven can accelerate the convergence rate, see Parter (1988), Evans and Yousif (1990) and Zhang (1996). Therefore, in this paper, we study on each scheme applied by each of the three ordering strategies.



(a). (b). (c).

Figure 3. (a), (b) and (c) show the ordering strategies for NA, RB and 4C, respectively in case of m= 7 at any level $t > 0$.

NUMERICAL EXPERIMENTS

In term of the computational implementation on the FSSOR, HSSOR and QSSOR methods, we only consider node points of type ● as shown in Figure 1 until specified iterative convergence is satisfied. Thus, the direct method (Abdullah, 1991; Jumat *et al.*, 1998; Othman & Abdullah, 2000, 2001) will be performed to obtain approximate solutions for the remaining points by using equations (3) and (4).

To review of the efficiency of all ordering strategies applied into the FSSOR, HSSOR and QSSOR methods, we conducted numerical experiments, which have done onto the two-dimensional unsteady convection-diffusion equation defined as (Harbans & Abdul Rahman, 1996)

$$\frac{\partial U}{\partial t} - v\left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2}\right) + w\left(\frac{\partial U}{\partial x} + \frac{\partial U}{\partial y}\right) = 0, \quad 0 \le x,\, y \le 1, \quad 0 < t < T \tag{5}$$

where v and w are coefficients. Then Dirichlet boundary conditions, initial condition, and the exact solution of problem (5) are given by

$$U(x, y, t) = \frac{1}{(4t+1)} \exp\left(-\frac{(x - wt - x_0)^2}{v(4t+1)} - \frac{(y - wt - y_0)^2}{v(4t+1)}\right), \quad 0 \le x,\, y \le 1, \quad 0 < t < T, \quad x_0 = y_0 = 0 .$$

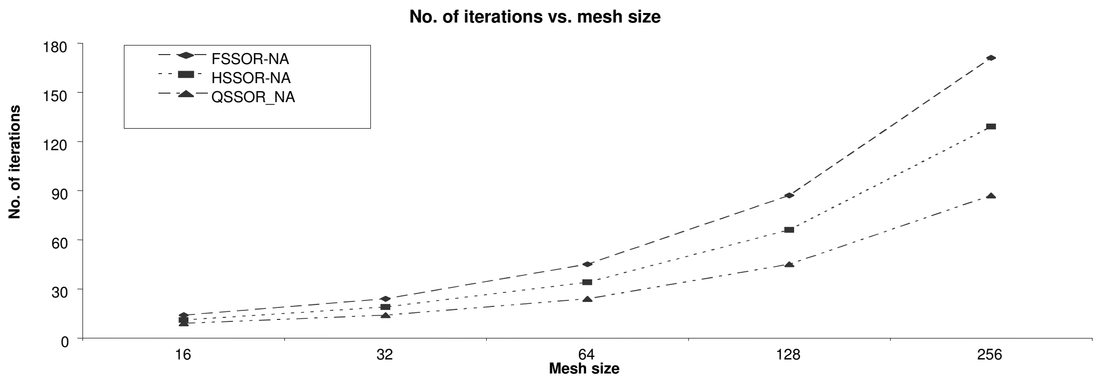


Figure 4. Number of iterations versus mesh size of the FSSOR, HSSOR and QSSOR methods using the NA strategy.

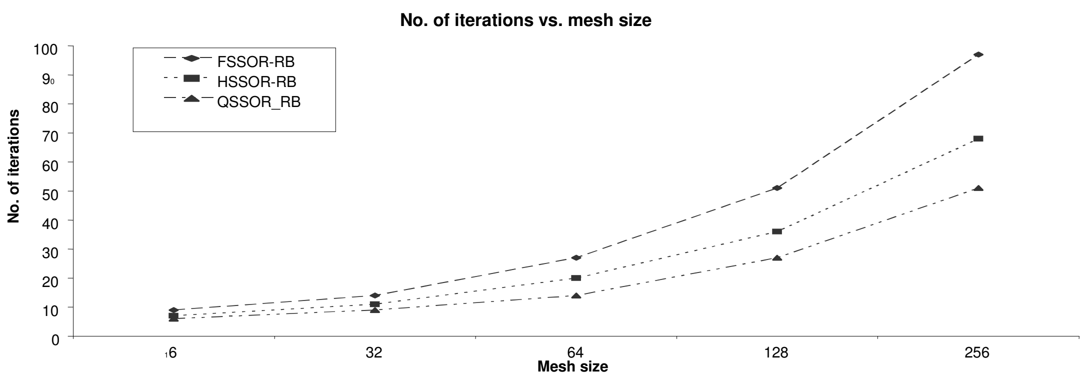


Figure 5. Number of iterations versus mesh size of the FSSOR, HSSOR and QSSOR methods using the RB strategy.

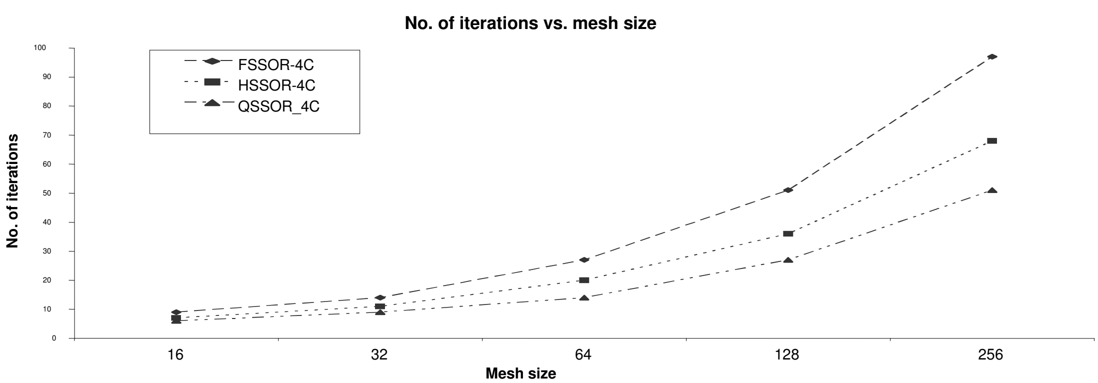


Figure 6. Number of iterations versus mesh size of the FSSOR, HSSOR and QSSOR methods using the 4C strategy.

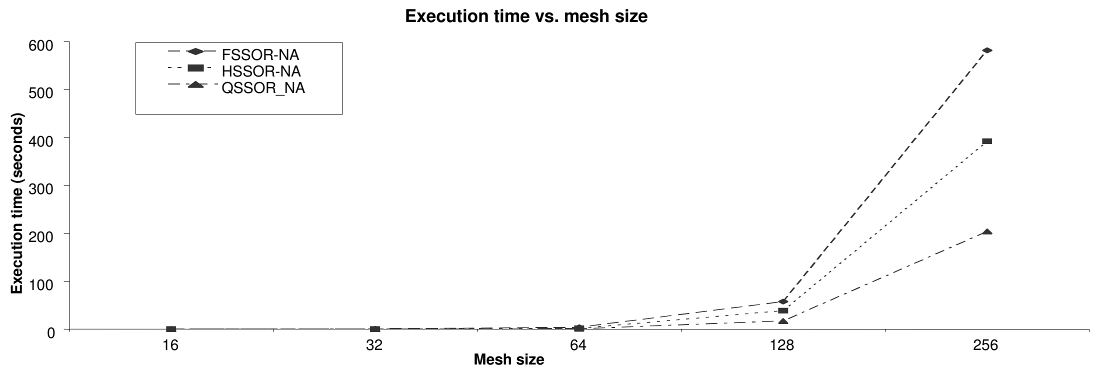


Figure 7. The execution time (seconds) versus mesh size of the FSSOR, HSSOR and QSSOR methods using the NA strategy.

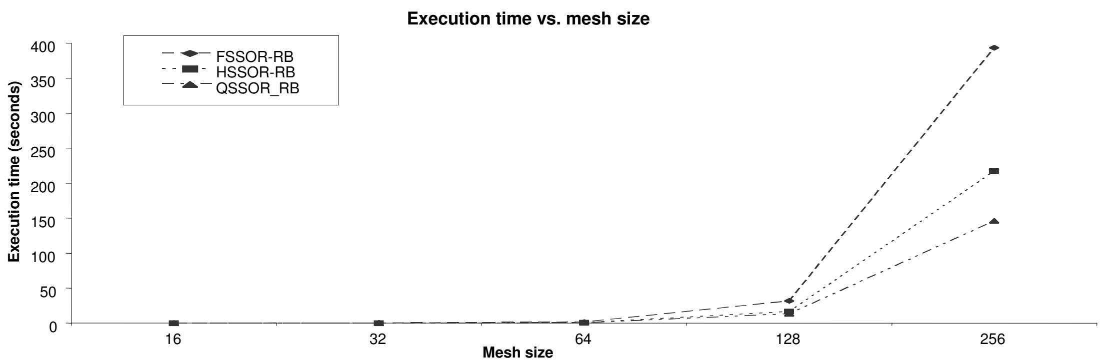


Figure 8. The execution time (seconds) versus mesh size of the FSSOR, HSSOR and QSSOR methods using the RB strategy.

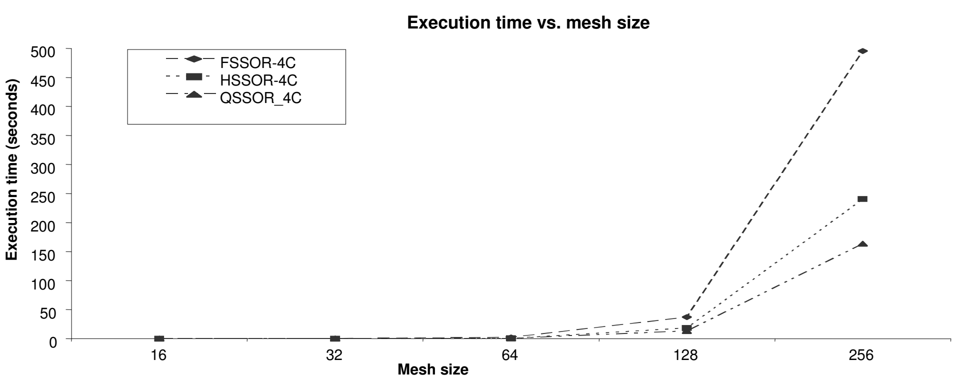


Figure 9. The execution time (seconds) versus mesh size of the FSSOR, HSSOR and QSSOR methods using the 4C strategy.

All results of numerical experiments, obtained from implementation of the FSSOR, HSSOR and QSSOR methods, have been recorded in Table 1. In the implementation mentioned above, the convergence test considered the tolerance error $\varepsilon = 10^{-10}$. Figures 4, 5 and **6** and Figures **7, 8** and 9 show number of iterations and the execution time versus grid size respectively.

CONCLUSION

According to the numerical results obtained and recorded in Table 1, the finding in Figures 4, 5, and 6 and Figures 7, 8, and 9 especially based on the RB strategy shows that a number of iterations and the execution time for the QSSOR have declined by 14.29 – 30.00% and 0.00 - 54.55% respectively compared with the HSSOR method. Overall, the numerical results shows that the QSSOR method is superior among all iterative methods in terms of a number of iterations and the execution time. This is because the computational complexity of the QSSOR method is quarter of the FSSOR method, while the HSSOR method is half. Even though number of iterations for each mesh size of the RB and 4C strategies with the QSSOR are the same, the RB strategy is the optimal ordering among all strategies. This is because of the execution time at m = 256 has declined by 10.61% compared to the 4C strategy.

Table 1. *Comparison of number of iterations, the execution time (seconds) and maximum absolute errors for the implementation of the respective NA, RB and 4C ordering strategies onto the FSSOR, HSSOR and QSSOR methods.*

No. of iterations

Ordering	FSSOR					HSSOR					QSSOR				
	16	32	64	128	256	16	32	64	128	256	16	32	64	128	256
NA	14	24	45	87	171	11	19	34	66	129	9	14	24	45	87
RB	9	14	27	51	97	7	11	20	36	68	6	9	14	27	51
4C	9	14	27	51	97	7	11	20	36	68	6	9	14	27	51

Execution time (seconds)

Ordering	FSSOR					HSSOR					QSSOR				
	16	32	64	128	256	16	32	64	128	256	16	32	64	128	256
NA	0.11	0.49	3.57	57.50	582.00	0.05	0.16	1.26	38.39	392.11	0.05	0.16	0.60	17.08	203.72
RB	0.06	0.28	1.65	31.80	393.32	0.05	0.11	0.66	16.64	217.23	0.05	0.05	0.43	13.35	146.16
4C	0.06	0.27	2.25	36.96	495.53	0.05	0.11	0.66	18.29	240.47	0.05	0.06	0.55	13.35	163.51

Maximum absolute errors

Ordering	FSSOR					HSSOR					QSSOR				
	16	32	64	128	256	16	32	64	128	256	16	32	64	128	256
NA	2.14 e-4	5.32 e-5	1.28 e-5	2.68 e-6	1.44 e-7	7.53 e-4	5.25 e-4	4.73 e-4	4.61 e-4	4.58 e-4	8.55 e-4	2.14 e-4	5.32 e-5	1.28 e-5	2.68 e-6
RB	2.14 e-4	5.32 e-5	1.28 e-5	2.69 e-6	1.56 e-7	7.53 e-4	5.25 e-4	4.73 e-4	4.61 e-4	4.58 e-4	8.55 e-4	2.14 e-4	5.32 e-5	1.28 e-5	2.69 e-6
4C	2.14 e-4	5.32 e-5	1.28 e-5	2.69 e-6	1.56 e-7	7.53 e-4	5.25 e-4	4.73 e-4	4.61 e-4	4.58 e-4	8.55 e-4	2.14 e-4	5.32 e-5	1.28 e-5	2.69 e-6

References

1. Abdullah, A. R. The Four Point Explicit Decoupled Group (EDG) method. A Fast Poisson Solver, *Intern. Journal. Computer Mathematics,* **34** (1991), 61-70.
2. Dahlquist, G. and Bjork, A. *Numerical methods,* Ned Anderson, New Jersey: Prentice Hall, 1974.
3. Evan, D. J. and Yousif, W. F. The Explicit Block Relaxation method as a grid smoother in the Multigrid V-cycle scheme, *Intern. Journal. Computer Mathematics,* **34** (1990), 71-78.
4. Harbans Kaur and Abdul Rahman Abdullah. Beberapa penyelesaian berangka bagi persamaan resapan-olakan dalam ruang dua dimensi, *Sains Malaysiana,* **25(1)** (1996),127-144.
5. Jumat Bin Sulaiman and Abdul Rahman Bin Abdullah. Kaedah lelaran kumpulan tak tersirat dengan penghampiran beza terhingga peringkat tinggi bagi persamaan Poisson, *Matematika,* **14** (1998), 27-37.

6. Jumat Sulaiman, Mohamed Othman and Abdul Rahman Abdullah. Skema kaedah lelaran kumpulan tak tersirat terubahsuai bagi persamaan resapan satu matra, *Borneo Science,* **4** (1998), 57-66.
7. Othman, M. B. and Abdullah, A. R. An Efficient Parallel Quarter-Sweep Point Iterative Algorithm for Solving Poisson Equation on SMP Parallel Computer, *Pertanika Journal of Science and Technology*, **8**(2) (2000), 161-174.
8. Othman. M. B and Abdullah, A. R. A New Technique Point Iterative Method for Solving Poisson Equation on MIMD Parallel Computer System, *Sains Malaysiana*, **27** (2001), 107-117.
9. Parter, S. V. Estimates for Multigrid methods based on Red Black Gauss-Seidel smoothers, *Numerical Mathematics,* **52** (1988), 701-723.
10. Zhang, J. Acceleration of Five Points Red Black Gauss-Seidel in Multigrid for Poisson Equations, *Applied Mathematics and Computation,* **80**(1) (1996), 71-78.