SOLVING THE TIME FRACTIONAL DIFFUSION EQUATIONS BY THE HALF-SWEEP SOR ITERATIVE METHOD

A. Sunarto
Faculty of Science and Natural Resources,
Universiti Malaysia Sabah,
88400 Kota Kinabalu, Sabah, Malaysia
andang99@gmail.com

J. Sulaiman
Faculty of Science and Natural Resources,
Universiti Malaysia Sabah,
88400 Kota Kinabalu,Sabah, Malaysia
jumat@ums.edu.my

A. Saudi
Faculty of Computing and Informatics,
Universiti Malaysia Sabah,
88400 Kota Kinabalu,Sabah, Malaysia
azali60@gmail.com

***Abstract*—This paper examines the effectiveness of the combination between Successive Over-Relaxation (SOR) iterative method with the Half-Sweep (HS) iteration namely Half-Sweep Successive Over-Relaxation (HSSOR) to solve one-dimensional time fractional diffusion equation numerically. To do this, the problem will be discretized to construct the Half-Sweep finite difference approximation equation via the Caputo's time fractional derivative and implicit finite difference discretization scheme. Then a linear system will be generated by this Half-Sweep approximation equation. Next the resulting of the linear system has been solved using HSSOR iterative method in which its effectiveness will be compared with the existing Successive Over-Relaxation method (known as Full-Sweep Gauss-Seidel (FSSOR)). One example is include to examine the effectiveness the proposed method. The findings of this study point out that the HSSOR iterative method is superior in term of number of iterations and computational time as compared with the FSSOR method.**

***Keywords*—*Caputo's fractional derivative; Implicit finite difference; HSSOR method*

I. INTRODUCTION

Many scientific problems in mathematics, physics, engineering and chemistry can be governed by using fractional partial differential equations (FPDEs) [1,2,3,4,5]. In fact, many proposed numerical techniques have been initiated to obtain a numerical and/or analytical solutions of the fractional problems. For instance, we have transform methods [6], finite elements together with the method of lines [3], explicit and implicit finite difference methods [3,7]. As mentioned in the finite difference methods, the explicit discretization scheme are categorised as a conditionally stable scheme whereas others discussions of finite difference schemes are can be found in the literature [8].

In this paper, we consider the finite difference methods to derive a finite difference approximation equation in representing the fractional problems. Basically, the approaches based on the finite difference methods can be divided into three categories: explicit, implicit and semi-explicit discretization schemes. Due the stability of the discretization process, we discretize the time fractional diffusion equations (TFDE) problem by using Half-Sweep implicit finite difference discretization scheme and Caputo fractional operator in order to derive the Half-Sweep Caputo's implicit approximation equation. Then this Half-Sweep approximation equation leads a linear system at each time level. Consequently, the characteristics of its coefficient matrix are sparse and large scale. To get numerical solutions over any linear systems iteratively, actually, many concepts of iterative methods have been initiated and enlightened by many researchers such as Young [9], Hackbusch [10], Saad [11], Evans [13] and Yousif and Evans [14,23]. in addition to that, it can be observed that several families of block iterative methods have been discussed by Evans [13], Ibrahim and Abdullah [12], Yousif and Evans [14,23] to show the advantages in term of its computation cost.

Besides the above iterative methods, Abdullah [24] has suggested the concept of iteration Half-Sweep iteration in which this iterative method is widely used to solve any linear systems. As known in previous studies, the advantage of the half-sweep iteration is to reduce the computational cost of the original linear system. In addition to that, the half-sweep iteration has been extensively extended by many researchers; see Ibrahim and Abdulah [12], Othman and Abdullah [21], Suliaman, Hasan and Othman.[22], Aruchunan and Sulaiman [17,18,19], Muthuvalu and Sulaiman [16], Yousif and Evans [14,23], Saudi and Sulaiman [25,26,27,28], Fauzi and

978-1-4799-5100-0/14/$31.00 ©2014 IEEE

Sulaiman [29,30], Khatim and Yit Hoe [20], Sulaiman and Akhir [31,32].

Due to the fast convergence rate of the point iteration, the standard SOR iterative method namely the Full-Sweep SOR (FSSOR) iterative method will be used as linear solver. By imposing the half-sweep iteration over the SOR method, we examine the effectiveness of the HSSOR iterative method for getting the numerical solutions of time-fractional parabolic partial differential equations (TPPDE's) by using the Half-Sweep Caputo's implicit approximation equation. To investigate the capability of the HSSOR method, the FSSOR iterative method acts as a control method.

To show the ability of this HSSOR method, let us consider time-fractional diffusion equation (TFDE's) be defined as

$$\frac{\partial^{\alpha} U(x,t)}{\partial^{\alpha}} = a(x)\frac{\partial^2 U(x,t)}{\partial x^2} + b(x)\frac{\partial U(x,t)}{\partial x} + c(x)U(x,t) \quad (1)$$

in which *a(x), b(x)* and *c(x)* can be considered known functions or constants. Then α is a parameter which represent to the fractional order of time derivative.

The outline of this paper is organized in the following way: In Section 2, the Half-Sweep Caputo's implicit approximation equation is derived. Then Section 3 deals with derivation of family of SOR methods. In Section 4, the formulation and implementation of the HSSOR iterative method are also presented. In Section 5 conducts numerical experiments and its numerical results and conclusion are included in Section 6.

II. Preliminaries

Before gaining the Half-Sweep Caputo's implicit approximation equation of the fractional problem (1), the following are some basic definitions of fractional derivative theory :

Definition 1. [8] The Riemann-Liouville fractional integral operator , J^{α} of order-α is defined as

$$J^{\alpha} f(x) = \frac{1}{\Gamma(\alpha)}\int_0^x (x-t)^{\alpha} f(t)dt, \alpha > 0, x > 0 \quad (2)$$

Definition 2.[8] The Caputo's fractional partial derivative operator, D^{α} of order -α is defined as

$$D^{\alpha} f(x) = \frac{1}{\Gamma(m-\alpha)}\int_0^x \frac{f^{(m)}(t)}{(x-t)^{\alpha-m+1}} dt, \alpha > 0 \quad (3)$$

with $m-1 < \alpha \le m$, m∈ N, $x > 0$

To get numerical solutions of the fractional problem (1) via the Half-Sweep implicit finite difference discretization scheme by using the finite difference method, firstly let the solution domain of the fractional problem be defined $0 \le x \le \gamma$, with $0 < \alpha < 1$, whereas the parameter α indicates to the fractional order of time derivative. Also consider the boundary conditions of fractional problem (1) be stated as

$$U(0,t) = g_0(t), U(\ell,t) = g_1(t),$$

whereas the initial condition is defined as

$$U(x,0) = g(x),$$

where $g_0(t)$, $g_1(t)$, and $g_2(x)$, are given functions and/or constants. In order to derive the approximation equation to the time fractional derivative in Eq. (1), firstly, consider Caputo's fractional partial derivative of order α , be stated as [8,9]

$$\frac{\partial^{\alpha} U(x,t)}{\partial t^{\alpha}} = \frac{1}{\Gamma(n-1)}\int_0^{\infty} \frac{\partial u(x-s)}{\partial t}(t-s)^{-\alpha} ds, \; t>0, \; 0<\alpha<1 \quad (4)$$

By using the Caputo's fractional operator in Eq.(4) and the Half-Sweep implicit finite difference discretization scheme, we can derive the Half-Swep Caputo's approximation equation. Actually, this Half-Sweep implicit approximation equation can be classified as unconditionally stable scheme. Further explanation on how to derive this approximation equation will be elaborated in the following section.

III. Half-sweep Caputo's Implicit Approximation

Before constructing the Half-Sweep Caputo's implicit approximation equation, let the Caputo's fractional partial derivative in Eq.(4) be rewritten as

$$D_t^{\alpha} U_{i,n} \cong \sigma_{\alpha,k} \sum_{j=1}^{n} \omega_j^{(\alpha)} \left(U_{i,n-j+1} - U_{i,n-j}\right) \quad (5)$$

and we have the following expressions

$$\sigma_{\alpha,k} = \frac{1}{\Gamma(1-\alpha)(1-\alpha)k^{\alpha}}$$

and

$$\omega_j^{(\alpha)} = j^{1-\alpha} - (j-1)^{1-\alpha}.$$

To facilitate us in discretizing the fractional problem (1) via the implicit finite difference discretization scheme, Firstly, its the solution domain of the fractional problem is divided uniformly into several subintervals. To do this, we consider several positive integers *m* and *n* in which the sizes of subinterval over space and time directions are stated as $h = \Delta x = \frac{\gamma - 0}{m}$ and $k = \Delta t = \frac{T}{n}$ respectively. According to these sizes of subinterval, we develop the uniformly finite grid network in the solution domain where the grid points over the space interval $[0,\gamma]$ and time interval $[0,T]$ are denoted $x_i = ih, \quad i = 0,1,2,...,m$ and $t_j = jk, \quad j = 0,1,2,...,n$ respectively. Then approximate values of the function $U(x,t)$ at the grid points are labeled as $U_{i,j} = U(x_i, t_j)$. By considering Eq. (5) and the Half-Sweep implicit finite difference discretization scheme, the Half-Sweep Caputo's implicit approximation equation of fractional problem (1) to the reference grid point at $(x_i, t_j) = (ih, jk)$ is can be shown as

$$\sigma_{\alpha,k}\sum_{j=1}^{n}\omega_j^{(\alpha)}\left(U_{i,n-j+1}-U_{i,n-j}\right)$$
$$=a_i\frac{1}{4h^2}\left(U_{i-2,n}-2U_{i,n}+U_{i+2,n}\right) \quad (6)$$
$$+b_i\frac{1}{4h}\left(U_{i+2,n}-U_{i-2,n}\right)+c_iU_{i,n},$$

for $i=2,4,...,m-2$. Actually, the approximation equation (6) is also categories as one of family of fully implicit finite difference approximation equations. Also this equation is first order accuracy in time direction and second order in space direction. For simplicity, consider the Half-Sweep approximation equation (6) be rewritten at the specified time level. For instance, we have for $n \geq 2$:

$$\sigma_{\alpha,k}\sum_{j=1}^{n}\omega_j^{(\alpha)}\left(U_{i,n-j+1}-U_{i,n-j}\right)=p_iU_{i-2,n}+q_iU_{i,n}+r_iU_{i+2,n}, \quad (7)$$

where

$$p_i=\frac{a_i}{4h^2}-\frac{b_i}{4h},$$
$$q_i=c_i-\frac{a_i}{2h^2},$$
$$r_i=\frac{a_i}{4h^2}+\frac{b_i}{4h}.$$

Also, we get for $n = 1$,

$$-p_iU_{i-2,1}+q_i^*U_{i,1}-r_iU_{i+2,1}=f_{i,0}, \quad i=2,4,...,m-2 \quad (8)$$

where

$$\omega_j^{(\alpha)}=1,$$
$$q_i^*=\sigma_{\alpha,k}-q_i,$$
$$f_{i,0}=\sigma_{\alpha,k}U_{i,0}.$$

Referring to Eq. (8), we can get the tridiagonal linear system that can be constructed in matrix form as

$$A\underset{\sim}{U}=\underset{\sim}{f} \quad (9)$$

where

$$A=\begin{bmatrix} q_2^* & -r_2 & & & & \\ -p_4 & q_4^* & -r_4 & & & \\ & -p_6 & q_6^* & -r_6 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -p_{m-2} & q_{m-2}^* & -r_{m-2} \\ & & & & -p_{m-2} & q_{m-2}^* \end{bmatrix}_{\left(\left(\frac{m}{2}\right)-1\right)x\left(\left(\frac{m}{2}\right)-1\right)},$$

$$\underset{\sim}{U}=\left[U_{2,1} \quad U_{4,1} \quad U_{6,1} \quad \cdots \quad U_{m-4,1} \quad U_{m-2,1}\right]^T,$$

$$\underset{\sim}{f}=\left[U_{21}+p_1U_{01} \quad U_{41} \quad U_{61} \quad \cdots \quad U_{m-4,1} \quad U_{m-2,1}+p_{m-2}U_{m,1}\right]^T.$$

IV. Formulation of Half-Sweep Successive Over Relaxation

To solve the tridiagonal linear system (9) iteratively, it can be concluded that the characteristic of its coefficient matrix (9) is large scale and sparse. In this paper, we consider the application of HSSOR method for solving linear system (9). As we know, the HSSOR method is essentially derived from the combination of FSSOR method and the half-sweep iteration. Due to the advantage of the Half-Sweep iteration and applying HSSOR method, let the tridiagonal linear system (9) be stated as

$$A=D-L-V \quad (10)$$

in which D, L and V are define as diagonal, lower triangular and upper triangular matrices respectively.

From the definition in Eq.(10), the general formulation of FSSOR and HSSOR iterative methods can be defined as

$$\underset{\sim}{U}^{(k+1)}=\left(D-\omega L\right)^{-1}\left(\left[(1-\omega)D+V\omega\right]\underset{\sim}{U}^{(k)}+\omega\underset{\sim}{f}\right) \quad (11)$$

where $\underset{\sim}{U}^{(k)}$ denotes an unknown vector at k[th] iteration. Based on Eq.(10), the implementation numerical experiments over HSSOR method can be stated in Algorithm 1.

Algorithm 1: HSSOR

i. Initialize all the parameters. Set $k=0$.
ii. For $j=1,2,\ldots,n-1,n$ and $i=2,4,,\ldots,m-4,m-2$
Calculate
$$\underset{\sim}{U}^{(k+1)}\leftarrow\left(D-\omega L\right)^{-1}\left(\left[(1-\omega)D+V\omega\right]\underset{\sim}{U}^{(k)}+\omega\underset{\sim}{f}\right)$$
iii. Check convergence test. If the convergence criterion i.e. $\left\|\underset{\sim}{U}^{(k+1)}-\underset{\sim}{U}^{(k)}\right\|\leq\varepsilon=10^{-10}$ is fulfilled, move step (iv). Otherwise go back to Step (ii).
iv. Display approximate solutions.

V. Numerical Experiment

In this experiment, we exemplify one tes example of the time fractional problem being used to demonstrate the accuracy and effectiveness of the HSSOR method as compared with the FSSOR method. For comparison purpose, three criteria have been considered such as number of iterations (K), computational time (in seconds) and maximum absolute error (MAE) at three different values of order α = 0.25, 0.50 and 0.75. For implementation of these two iterative schemes, the convergence test considered the tolerance error, which is fixed as $\varepsilon=10^{-10}$.

To indicate the effectiveness of HSSOR iteration method, consider one example for family of time fractional problem (1) be given as [15]

$$\frac{\partial^{\alpha}U(x,t)}{\partial t^{\alpha}}=\frac{\partial^{2}U(x,t)}{\partial x^{2}},\quad 0<\alpha\leq 1, 0\leq x\leq\gamma,\quad t>0 \quad (12)$$

subjected to the following boundary conditions, which are stated as

$$U(0,t)=\frac{2kt^{\alpha}}{\Gamma(\alpha+1)},\ U(\ell,t)=\ell^{2}+\frac{2kt^{\alpha}}{\Gamma(\alpha+1)},$$

Then its initial condition is given as follow :

$$U(x,0)=x^{2}.$$

As taking $\alpha=1$, it can be seen that problem (12) can be reduced to the standard diffusion equation

$$\frac{\partial U(x,t)}{\partial t}=\frac{\partial^{2}U(x,t)}{\partial x^{2}},\quad 0\leq x\leq\gamma,\quad t>0, \quad (13)$$

where the initial condition is given as

$$U(x,0)=x^{2},$$

and boundary conditions as

$$U(0,t)=2kt,\qquad U(\ell,t)=\ell^{2}+2kt,$$

Then exact solution of Problem (12) can be expressed as

$$U(x,t)=x^{2}+2kt.$$

Now by applying the series

$$U(x,t)=\sum_{n=0}^{m-1}\frac{\partial^{n}U(x,0)}{\partial t^{n}}\frac{t^{n}}{n!}+\sum_{n=1}^{\infty}\sum_{i=0}^{m-1}\frac{\partial^{mn+i}U(x,0)}{\partial t^{mn+i}}\frac{t^{n\alpha+i}}{\Gamma(n\alpha+i+1)}$$

to $U(x,t)$ for $0<\alpha\leq 1,$ it can be pointed out that the exact solution of Problem (11) can be stated as

$$U(x,t)=x^{2}+2k\frac{t^{\alpha}}{\Gamma(\alpha+1)}.$$

Based on three difference values of α, the numerical results obtained from implementations of FSSOR and HSSOR iterative methods over problem (12), have been tabulated in Table 1 for different values of subinterval sizes, *m* = 128, 256, 512, 1024, and 2048.

Based on Table 1, the reduction percentage of computational time, R_m is given as

$$R_{m}=\frac{\eta_{m}-\psi_{m}}{\eta_{m}}\times 100$$

where ψ_m and η_m are computational time of HSSOR and FSSOR methods respectively. Then the reduction percentage of number of iterations, R_b is defined as

$$R_{b}=\frac{\eta_{b}-\psi_{b}}{\eta_{b}}\times 100$$

where ψ_b and η_b are number of iterations of HSSOR and FSSOR methods respectively.

VI. CONCLUSION

As a conclusion for the numerical solution of the time fractional diffusion problems, this paper deals with the implementation of HSSOR iterative method to solve a linear system generated by the Half-Sweep Caputo's implicit approximation equations Through numerical experiments results from Table 2 by comparing the performance between the FSSOR and HSSOR iterative methods at three different values of $\alpha=0.25, 0.50$ and 0.75, it can be seen that the percentage reduction of number of iterations for the HSSOR iterative method have declined approximately by 2.24-76.58%, 0.71-80.53%, and 2.41-75.45% respectively as compared with the FSSOR method. In fact, implementations of computational time for HSSOR method are much faster about 32.21-84.99%, 29.64-86.35%, and 31.03-84.99% respectively than the FSSOR method. It can be conclude that the HSSOR method involves less number of iterations and computational time as compared with FSSOR methods. According to the accuracy of FSSOR and HSSOR iterative methods, it can be stated that the numerical solutions of both method are in good agreement. Besides FSOR and HSSOR iterative methods, the family of AOR methods, see Hadjidimos, Hallett, Sunarto *et al* [33,34,35] which are categorized as one of of the single step iterative family, can be considered for future work. In addition to those iterative families, two-step iterative family (Ruggiero & Galligani [36]; Galligani [37,38]; Evans & Sahimi [39]; Sahimi *et al.* [40]; Sulaiman *et al.* [41]) are also interesting to be examined for the proposed problem.

REFERENCES

[1] Mainardi, F,, Fractals and Fractional Calculus Continuum Mechanics. Heidelberg: Springer-Verlag, 1997, pp 291-348.

[2] Diethelm, K. and Freed, A.D, On the solution of nonlinear fractional order differential equation used in the modeling of viscoelasticity, in Scientific Computing in Chemical engineering II Computational Fluid Dynamic, Reaction Engineering and Molecular Properties, Heidelberg: Springer-Verlag, 1999, pp 217-224.

[3] Meerschaert, M.M. and Tadjeran, C, 2004, Finite difference approximation for fractional advection-dispersion flow equations, *JCAM*, 172, 2004, pp 145-155.

[4] Chaves, A, Fractional diffusion equation to describe Levy flight, *Phys.Lett, A*239, 1998, pp 13-16.

[5] Agrawal, O.P, Solution for a fractional diffusion-wave equation defined in a bounded domain, *Nonlinear Dynamic*, 29, 2002 , pp 145-155

[6] Yuste, S.B. and Acedo, L,, An explicit finite difference method and a new von Neumann-type stability analysis for fractional diffusion equations, *SIAM J. Numer. Anal.*42 (5) , 2005, pp 1862-1874.

[7] Yuste, S.B, Weighted average finite difference method for fractional diffusion equations, *J.Comp.Phys*.216, 2006, pp 264-274.

[8] Zhang, Y, A Finite Difference Method For Fractional Partial Differential Equation, *Applied Mathematics And Computation 215*, 2009, pp 524-529.

[9] Young, D.M, Iterative solution of large linear systems. London: Academic Press, 1971.

[10] Hackbusch, W, Iterative solution of large sparse systems of equations. New York: Springer-Verlag, 1995.

[11] Saad, Y, Iterative method for sparse linear systems. Boston: International Thomas Publishing, 1996.

[12] Ibrahim, A and Abdullah, A.R, Solving the two dimensional diffusion equation by the Four Point Explicit Decoupled Group (EDG) iterative method. *International Journal Computer Mathematics*. 58, 1995, pp 253-256.

[13] D.J. Evans.Group Explicit Iterative methods for solving large linear systems, *Int. J. Computer Maths., 1985,* **17**: pp 81-108.

[14] Yousif, W. S and Evans, D.J, Explicit Group Iterative Methods for solving elliptic partial differential equations in 3-space dimensions. *Int. J. Computer Maths.,*18 , 1986, pp 323-340.

[15] Ali, S E, Ozgur, B and Korkmaz, E, Analysis Of Fractional Partial Differential Equations By Taylor Series Expansion. Boundary Value Problem, A Springer Open, 2013, pp 68.

[16] Muthuvalu, M.S and Sulaiman, J, Computational Methods Based on Complexity Reduction Approach for First ind Linear Fredholm Integral Equations with Semi-Smooth Kernel*, Jurnal Of Modern in Numeric Mathematics,* 2012, pp 64-71.

[17] Aruchunan, E and Sulaiman, J, Half-Sweep Conjugate Gardient method for Solving First Order Linear Fredholm Integro-differential Equations, *Australian Journal of Basic and applied Sciences*, 2011, pp 38-43.

[18] Aruchunan, E and Sulaman, J. Half–Sweep Quadrature-Difference Schemes with Iterative Method In Solving Linear Fredholm Integro-Differential Equations, *Progress in Applied Mathematics*, 2013, pp 11-21.

[19] Aruchunan, E and Sulaiman, J. Applications of the Central-Difference with Half-Sweep Gauss-Seidel Method for Solving First Order Linear Fredholm Integro-Differential Equations, *World Academy of Science, Engineering and Technology,*, 2012, pp 335-339.

[20] Hasan, M.K, and YitHoe, N, Impact of Alternating Left Right Strategy on Half-Swep Succesive Over Relaxation for The Solution of Two Dimensional Boundary value problems, Australian Journal of Basic and applied Sciences, 2013, pp 391-398.

[21] Othman, M and Abdullah, A.R , The Half-Sweep Multigrid Method as a fast Multigrid Poisson Solver*, Int J.Comp Math,,* 1998, pp 319-329.

[22] Sulaiman, J, Hasan, M.K and Othman, M, The Half-Sweep Iterative Alternating Decomposition Explicit (HSIADE) Method for Diffusion Equations, *In Comp.l and Info Science,* 2004.

[23] Yousif, W and Evans, D.J, Explicit De-coupled Group Iterative Methods and Their Implementations, Parallel Algorithm and Applications, 1995 , pp 53-71.

[24] Abdulah, A.R, The Four Point Explicit Decoupled group (EDG) Method: A Fast Poisson Solver, *Int.J Comp.Math ,* 1991, pp 61-70.

[25] Saudi, A and Sulaiman, J, Path Planning for Mobile Robot with Half-Sweep Successive Over-Relaxations (HSSOR) Iterative Method, Symposium on Progress in Information&Communication Technology, 2009, pp 57-61.

[26] Saudi, A and Sulaiman, J, Roboth Path Planning Using Laplacian Behaviour-Based Control (LBBC) via Half-Swep SOR, Proceeding IEEE ISBN:978-1-4673-5613-8, 2013, pp 424-429.

[27] Saudi, A and Sulaiman, J, Path Planning for Indoor mobile Robot Using Half-Sweep SOR via Nine-Point Laplacian (HSSOR9L), IOSR Journal of Mathematics, 2012, pp 01-07.

[28] Saudi, A and Sulaiman, J, Laplacian Behaviour-Based Control for Robot Path Planning Using Full-Sweep Successive Over-relaxations via Nine-Point Laplacian (FSSOR9L), International Journal of Apllied Science and Technology, 2012, pp 255-261.

[29] Fauzi, N.I.M and Sulaiman, J, Quarter-Sweep Modified SOR Iterative Algorithm and Cubic Spline Basis for The Solution od Second Order two-Point Boundary Value Problems, Journal of Applied Science, 2012, pp 1-8.

[30] Fauzi, N.I.M and Sulaiman, J, Half-Sweep Modified Successive Over Relaxation Method for Solving Second Order Two-Point Boundary Value Problems using Cubic Spline, Int.J.Cotemp.Math.Science, 2012, pp 1579-1589.

[31] Sulaiman,J et al, Numerical Solutions of Nonliniear Second –Order Two-Point Boundary Value Problems using Half-Sweep SOR with Newton Method, SST UMS, 2012.

[32] Akhir,M..K..M et al, Half-Sweep Successive Over Relaxations for Solving Two-Dimensional Helmholtz Equations, Australian Journal of Basic and applied Science, 2011, pp 3033-3039.

[33] Hadjidimos, A., Accelerated OverRelaxation Method. *Mathematics of Computation.*, 1978, 32**,** 149-157.

[34] Hallett, A.J.H, The Convergence Of Accelerated OverRelaxation Iterations, Mathematics of Computation. 1986 47 (125): 219: 223

[35] Sunarto,A, Sulaiman, J and Saudi, A. Implicit finite difference solution for time-fractional diffusion equations using AOR method. Journal of Physics: Conference Series, 2014 .495**:** 012032

[36] Ruggiero, V. & Galligani, E. 1990. An iterative method for large sparse systems on a vector computer, *Comput. Math. Appl. 20*(1):25-28.

[37] Galligani, E. 2003. The Newton-Arithmetic mean method for the solution of systems of nonlinear equations; *Applied Mathematics and Computation,* 134:9-34.

[38] Galligani, E. 2006. The Arithmetic Mean method for solving systems of nonlinear equations in finite differences; *Applied Mathematics and Computation,* 181:579-597.

[39] Evans, D.J. & Sahimi, M.S. 1988. The Alternating Group Explicit iterative method (AGE) to solve parabolic and hyperbolic partial

differential equations. *Ann. Rev. Num. Fluid Mechanic and Heat Trans*, 2:283-389.

[40] Sahimi, M.S., Ahmad, A. & Bakar, A.A. 1993. The iterative alternating decomposition explicit (IADE) method to solve the heat conduction equation, International. *Journal of Computer Mathematics,* 47:219–229.

[41] Sulaiman, J., Hasan, M.K. & Othman, M. 2004. The Half-Sweep Iterative Alternating Decomposition Explicit (HSIADE) method for diffusion equations. Lecture Note on Computer Science (LNCS 3314):57-63.

.

TABLE 1. Comparison between number of iterations (K), the computational time (seconds) and maximum absolute errors for both iterative methods at $\alpha = 0.25, 0.50, 0.75$

M	Method	α = 0.25			α = 0.50			α = 0.75		
		K	Time	MAE	K	Time	MAE	K	Time	MAE
128	FSSOR	714	2.08	9.95e-05	703	1.99	9.84e-05	705	2.03	1.29e-04
	HSSOR	698	1.41	9.95e-05	698	1.40	9.84e-05	688	1.40	1.29e-04
256	FSSOR	1461	6.90	9.95e-05	769	4.01	9.84e-05	769	4	1.29e-04
	HSSOR	714	2.17	9.95e-05	703	2.15	9.84e-05	705	2.19	1.29e-04
512	FSSOR	6239	55.97	9.96e-05	3951	35.08	9.84e-05	1821	16.51	1.29e0-4
	HSSOR	1461	8.4	9.95e-05	769	4.79	9.84e-05	769	4.81	1.29e-04
1024	FSSOR	23626	415.43	9.97e-05	15229	268.33	9.86e-05	7417	129.41	1.30e-04
	HSSOR	6239	65.78	9.96e-05	3951	41.56	9.84e-04	1821	19.42	1.29e-04
2048	FSSOR	87221	3204.74	1.00e-04	56530	2058.11	9.91e-05	27855	1006.92	1.30e-04
	HSSOR	23626	499.01	9.97e-05	15229	319.26	9.85e-05	7417	155.02	1.30e-04

TABLE 2. Calculation of the percentage reduction for number of iterations ($\mathbf{R_b}$) and the computational time ($\mathbf{R_m}$) for the HSSOR method compared with FSSOR method at $\alpha = 0.25, 0.50, 0.75$

M	α = 0.25		α = 0.50		α = 0.75	
	R_b	R_m	R_b	R_m	R_b	R_m
128	2.24	32.21	0.71	29.64	2.41	31.03
256	51.12	68.55	8.58	46.38	8.32	45.25
512	76.58	84.99	80.53	86.35	57.77	70.86
1024	73.59	84.16	74.05	84.51	75.45	84.99
2048	72.91	84.43	73.06	84.48	73.37	84.60