Malaysian Journal of Mathematical Sciences 13(S) April: 15–26 (2019)
Special Issue: 3rd International Conference on Mathematical Sciences and Statistics (ICMSS2018)

MALAYSIAN JOURNAL OF MATHEMATICAL SCIENCES

Journal homepage: http://einspem.upm.edu.my/journal

EGSOR Iterative Method for the Fourth-Order Solution of One-Dimensional Convection-Diffusion Equations

Muhiddin, F. A. *[1] and Sulaiman, J. [2]

[1] *Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Sabah Campus, Malaysia*
[2] *Mathematics with Economics Programme, Universiti Malaysia Sabah, Malaysia*

E-mail: fatihah.anas@uitm.edu.my
** Corresponding author*

ABSTRACT

In this paper, the effectiveness of two-point Explicit Group Successive Over-Relaxation (EGSOR) iterative method in obtaining the approximate solution of one-dimensional (1D) convection-diffusion equations with the fourth-order implicit finite difference scheme is investigated. For the fourth-order solution of the proposed problems, the combination of second-order and fourth-order implicit finite difference approximation equations have been used to derive the generated pentadiagonal linear system. For comparison purposes, other point iterative methods which are Gauss-Seidel (GS) and Successive Over-Relaxation (SOR) are also included as control methods. Three numerical examples have been considered to access the efficiency of the proposed iterative method. Finally, from the numerical results obtained, it can be concluded that the two-point EGSOR iterative method shows superiority in terms of number of iterations and execution time in comparison to the other iterative methods.

Keywords: Fourth-order finite difference, EGSOR iterative method, Convection-diffusion equations.

1. Introduction

Unsteady convection-diffusion equation is a parabolic type of partial differential equations in the case of unsteady transport equations. Various previous researchers have mentioned the importance of understanding the transport phenomena, especially in the industry and engineering fields. In finding the numerical solutions, recent researchers have actively shown interest in high-order schemes. For instance, Ge et al. (2018) discussed the high-order compact Alternating Direction Implicit (ADI) method to solve 3D unsteady convection-diffusion equation. Meanwhile, Dai et al. (2016) and Sun and Li (2014) discussed the numerical solution of 2D unsteady convection-diffusion equation using higher-order ADI with completed Richardson extrapolation method and with combined compact difference (CCD) scheme respectively.

In this paper, we focus on the following linear one-dimensional convection-diffusion equation with constant coefficients

$$\frac{\partial U}{\partial t} + \alpha \frac{\partial U}{\partial x} = \epsilon \frac{\partial^2 U}{\partial x^2}, \quad x \in [a,b], \quad 0 < t \leq T, \tag{1}$$

subject to the initial and boundary conditions

$$\begin{aligned} &U(x,0) = f(x), && \text{a} \leq x \leq b, \\ &U(a,t) = g_0(t), \quad U(b,t) = g_1(t), && 0 \leq x \leq T, \end{aligned}$$

where α is the convection parameter and ϵ is the diffusion parameter.

Discretization of problem (1) will produce large and sparse linear systems which is best to be solved using iterative methods. The early development of these methods received the attention of researchers, such as Young (1954, 1971, 1972, 1976), Hackbusch (1994) and Saad (2003). In fact, Young (1954) has introduced the Successive Over-Relaxation iterative method in order to improve the convergence rate of the classical iterative methods. Due to the advantage of the SOR point iterative method, Evans (1985) introduced the families of block iterative methods, which further discussed by Evans and Sahimi (1989) and Evans and Yousif (1990). Based on their findings on the efficiency of block iterative methods, this paper investigates the effectiveness of two-point EGSOR (2EGSOR) iterative method for solving problem (1) by using the fourth-order implicit approximation equations. To do this, let us discretize problem (1) by using the uniformly divided grid spacing which is assumed as

$$\Delta x = \tfrac{b-a}{m} = h, \quad m = n+1, \quad \Delta t = \tfrac{T-0}{M} \tag{2}$$

Figure 1 depicts the finite grid network that is used to guide us in the formulation of the GS and SOR iterative methods which are applied onto each interior node point until convergence is attained.

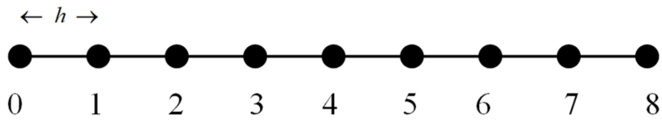

Figure 1: The distribution of uniform node points for the solution domain at m=8.

2. Formulation of Fourth-Order Implicit Approximation Equation

Before constructing a fourth-order finite difference approximation equation using implicit finite difference scheme, let us consider the second-order approximation equation for problem (1) in general form as

$$a_1 U_{i-1,j+1} + b_1 U_{i,j+1} + c_1 U_{i+1,j+1} = F_{i,j} \tag{3}$$

where

$$\mathrm{a}_1 = -(\gamma_0^* + \gamma_1^*), \quad \mathrm{b}_1 = 1 + 2\gamma_0^* + \gamma_2^*, \quad \mathrm{c}_1 = \gamma_1^* - \gamma_0^*, \quad F_{i,j} = U_{i,j}.$$

and

$$\gamma_0^* = \tfrac{\epsilon \Delta t}{h^2}, \quad \gamma_1^* = \tfrac{\alpha \Delta t}{2h}, \quad \gamma_2^* = \beta \Delta t.$$

Meanwhile, the fourth-order approximation equation has been expressed in general form as

$$a_2 U_{i-2,j+1} + b_2 U_{i-1,j+1} + c_2 U_{i,j+1} + d_2 U_{i+1,j+1} + e_2 U_{i+2,j+1} = F_{i,j} \tag{4}$$

where

$$\begin{array}{lll} \mathrm{a}_2 = \gamma_0 + \gamma_1, & \mathrm{b}_2 = -(8\gamma_1 + 16\gamma_0), & \mathrm{c}_2 = 1 + 30\gamma_0 + \gamma_2, \\ \mathrm{d}_2 = 8\gamma_1 - 16\gamma_0, & \mathrm{e}_2 = \gamma_0 - \gamma_1, & F_{i,j} = U_{i,j}. \end{array}$$

and

$$\gamma_0^* = \tfrac{\epsilon \Delta t}{12h^2}, \quad \gamma_1^* = \tfrac{\alpha \Delta t}{12h}, \quad \gamma_2^* = \beta \Delta t.$$

Equations (3) and (4) can be visualised clearly in the form of computational molecule as depicted in Figures 2 and 3 respectively.

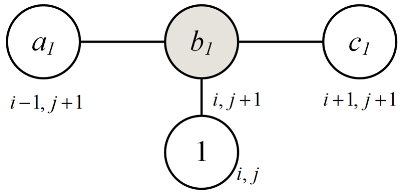

Figure 2: The second-order implicit computational molecule.

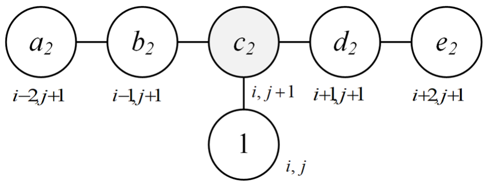

Figure 3: The fourth-order implicit computational molecule.

From Figures 2 and 3, illustrated clearly that the approximate values along nodes $i = 1$ and $i = m-1$ are impossible to be obtained using the fourth-order approximation equation (4) since some points are located in the exterior of the solution domain (1). To deal with this matter, we applied the second-order approximation equation (3) along the two node points. The combination of equations (3) and (4) will generate sets of linear systems in matrix form at any time level $j+1$ as

$$A\underline{\mathrm{U}}_{j+1} = \underline{\mathrm{F}}_j \tag{5}$$

where

$$\mathbf{A} = \begin{pmatrix} b_1 & c_1 & & & & & \\ b_2 & c_2 & d_2 & e_2 & & & \\ a_2 & b_2 & c_2 & d_2 & e_2 & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & a_2 & b_2 & c_2 & d_2 & e_2 \\ & & & a_2 & b_2 & c_2 & d_2 \\ & & & & & a_1 & b_1 \end{pmatrix}$$

$$\underline{\mathrm{U}}_{\mathbf{j+1}} = \begin{pmatrix} U_{1,j+1} \\ U_{2,j+1} \\ \vdots \\ U_{m-2,j+1} \\ U_{m-1,j+1} \end{pmatrix}, \underline{\mathrm{F}}_{\mathbf{j}} = \begin{pmatrix} F_{1,j} - a_1 U_{0,j+1} \\ F_{2,j} - a_2 U_{0,j+1} \\ \vdots \\ F_{m-2,j} - e_2 U_{m-1,j+1} \\ F_{m-1,j} - c_1 U_{m,j+1} \end{pmatrix}$$

3. Derivation of two-point EGSOR Iterative Method

In order to investigate the effectiveness of the 2EGSOR iterative method for solving problem (1) using the fourth-order finite difference approximation equation, the SOR and GS iterative methods are assigned as the reference methods.

3.1 Formulation of SOR Iterative Method

To derive the formulation of the SOR iterative method, we decomposed the coefficient matrix A in equation (5) as

$$A = L + D + V \tag{6}$$

where D is the diagonal, L is strictly lower triangular and V is strictly upper triangular matrices of matrix A respectively. From equation (6), therefore the SOR iterative scheme can generally be defined as follows Young (1954, 1971, 1972, 1976)

$$\underline{U}_j^{(k+1)} = (1-\omega)\underline{U}_j^{(k)} + \omega(D+L)^{-1}(\underline{F} - V\underline{U}_j^{(k)}) \tag{7}$$

where, ω represent the relaxation factor and $\underline{U}_j^{(k+1)}$ is the unknown vector at k^{th} iteration. From equation (7), it is noted that the SOR iterative method is equivalent to the GS iterative method at $\omega = 1$. Therefore, to ensure the family of SOR iterative method accelerate faster, the optimal value of ω should be set within the range of $1 \leq \omega < 2$.

Thus, the SOR algorithm to solve convection-diffusion in problem (1) is as summarized in Algorithm 1.

Algorithm 1. SOR algorithm

1. Initialized $\underline{U}_j^{(0)} \leftarrow 0$ and $\epsilon \leftarrow 10^{-10}$.
2. Assign the optimal value ω,
3. For $j = 1, 2, 3, ..., M$, implement:
   (a) Assign $\underline{U}_j^{(0)} \leftarrow 0$
   (b) Solve linear system (5) iteratively by using equation (7)

(c) Perform the convergence test, $| \underline{U}_j^{(k+1)} - \underline{U}_j^{(k)} | \leq \epsilon = 10^{-10}$. If yes, go to step (d). Otherwise, repeat step (b).

(d) Check time level, $j = M$. If yes, go to step (iv). Otherwise, repeat step (a).

4. Display approximate solutions.

3.2 Two-Point EGSOR Iterative Method

To derive the formulation of the two-point EGSOR iterative method, let us consider a group of two points as depicted in Figure 4.

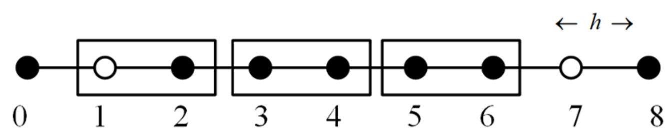

Figure 4: The distribution of uniform nodal points of 2EGSOR for m=8.

By taking similar finite grid network as in Figure 1, Figure 4 illustrates the application of 2EGSOR iterative method which is onto each two-point block until iterative convergence is achieved.

By considering second- and fourth-order finite difference approximation equations (3) and (4), for the first block, this method can be generally expressed as

$$\begin{bmatrix} b_1 & c_1 \\ b_2 & c_2 \end{bmatrix} \begin{bmatrix} U_{1,j+1} \\ U_{2,j+1} \end{bmatrix} = \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} \tag{8}$$

where

$$S_1 = F_{1,j} - a_1 U_{0,j+1}$$
$$S_2 = F_{2,j} - a_1 U_{0,j+1} - d_2 U_{3,j+1} - e_2 U_{4,j+1}$$

Meanwhile for $i = 3, 7, \ldots, m-3$, other remaining two-point blocks can be stated as

$$\begin{bmatrix} c_2 & d_2 \\ b_2 & c_2 \end{bmatrix} \begin{bmatrix} U_{i,j+1} \\ U_{i+1,j+1} \end{bmatrix} = \begin{bmatrix} S_3 \\ S_4 \end{bmatrix} \tag{9}$$

where

$$S_3 = F_{i,j} - a_2 U_{i-2,j+1} - b_2 U_{i-1,j+1} - e_2 U_{i+2,j+1}$$
$$S_4 = F_{i+1,j} - a_2 U_{i-1,j+1} - d_2 U_{i+2,j+1} - e_2 U_{i+3,j+1}$$

Now, by determining the inverse of the coefficient matrix in equations (8) and (9), the two-point EGSOR iterative method for the first block can be generally shown as

$$\begin{bmatrix} U_{1,j+1} \\ U_{2,j+1} \end{bmatrix}^{(k+1)} = \frac{\omega}{\mid B \mid} \begin{bmatrix} c_2 & -c_1 \\ -b_2 & b_1 \end{bmatrix} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} + (1-\omega) \begin{bmatrix} U_{1,j+1} \\ U_{2,j+1} \end{bmatrix}^{(k)} \tag{10}$$

where

$$\mid B \mid = c_2 b_1 - c_1 b_2.$$

Next, for the consecutive remaining blocks until $i = m-3$, can be generally stated as

$$\begin{bmatrix} U_{i,j+1} \\ U_{i+1,j+1} \end{bmatrix}^{(k+1)} = \frac{\omega}{\mid C \mid} \begin{bmatrix} c_2 & -d_2 \\ -b_2 & c_2 \end{bmatrix} \begin{bmatrix} S_3 \\ S_4 \end{bmatrix} + (1-\omega) \begin{bmatrix} U_{i,j+1} \\ U_{i+1,j+1} \end{bmatrix}^{(k)} \tag{11}$$

where

$$\mid C \mid = c_2^2 - b_2 d_2.$$

Thus, the 2EGSOR algorithm to solve convection-diffusion in problem (1) is as summarized in Algorithm 2.

Algorithm 2. 2EGSOR algorithm

1. Initialized $\underline{\mathrm{U}}_j^{(0)} \leftarrow 0$ and $\epsilon \leftarrow 10^{-10}$.
2. Assign the optimal value ω,
3. For $j = 1, 2, 3, ..., M$, implement:
   (a) Assign $\underline{\mathrm{U}}_j^{(0)} \leftarrow 0$
   (b) Solve linear system (5) iteratively by using equation (10) and (11)
   (c) Perform the convergence test, $\mid \underline{\mathrm{U}}_j^{(k+1)} - \underline{\mathrm{U}}_j^{(k)} \mid \leq \epsilon = 10^{-10}$. If yes, go to step (d). Otherwise, repeat step (b).
   (d) Check time level, $j = M$. If yes, go to step (iv). Otherwise, repeat step (a).
4. Display approximate solutions.

4. Numerical Experiments

In this section, we tested linear equations of problem (1) with constant coefficients to analyse the performance of the 2EGSOR iterative method over the other two iterative methods, namely SOR and GS iterative methods. Here, three types of convection-diffusion equations (i.e. heat equation, convection-dominated equations, and a convection-diffusion equation with similar coefficients) are presented in examples 1 to 3 respectively. Three factors which are the number of iterations, execution time (seconds), and maximum absolute error are measured as comparative analyses. For convergence test of the iterative methods, the tolerance error value is set to $\epsilon = 10^{-10}$.

Example 1: Consider the following heat equation (Sulaiman et al., 2010)

$$\frac{\partial U}{\partial t} + \alpha \frac{\partial U}{\partial x} = \epsilon \frac{\partial^2 U}{\partial x^2}, \quad x \in [0,1], \quad 0 \leq t \leq T,$$

with $\alpha = 0.0$, $\epsilon = 1.0$, and subject to the initial condition

$$U(x,0) = sin(\pi x) + 3sin(2\pi x).$$

the boundary condition and the exact solution to this problem is

$$U(x,t) = e^{-\pi^2 t} sin(\pi x) + 3e^{-4\pi^2 t} sin(2\pi x), \quad 0 \leq t \leq 1.$$

Example 2: Consider the following convection-dominated problem (Mittal and Jain, 2012)

$$\frac{\partial U}{\partial t} + \alpha \frac{\partial U}{\partial x} = \epsilon \frac{\partial^2 U}{\partial x^2}, \quad x \in [0,1], \quad 0 \leq t \leq T,$$

with $\alpha = 3.5$, $\epsilon = 0.022$, and subject to the initial condition

$$U(x,0) = exp(px).$$

The exact solution and boundary condition to this problem is

$$U(x,t) = exp(px + qt), \quad 0 \leq t \leq 1.$$

with p = 0.02854797991928 and $q = -0.0999$.

Example 3: Consider the following convection-diffusion equation (Mittal and Jain, 2012)

$$\frac{\partial U}{\partial t} + \alpha \frac{\partial U}{\partial x} = \epsilon \frac{\partial^2 U}{\partial x^2}, \quad x \in [0,1], \quad 0 \leq t \leq T,$$

with $\alpha = 0.1$, $\epsilon = 0.02$, and subject to the initial condition

$$U(x,0) = exp(px),$$

The exact solution and boundary condition to this problem is

$$U(x,t) = exp(px + qt), \quad 0 \le t \le 1.$$

with p = 1.1771243444677 and $q = -0.09$.

Then, the numerical results of the three examples of convection-diffusion problems are presented in Tables 1 to 3.

From the numerical experiment results in Tables 1 to 3, the percentage of improvement between SOR and 2EGSOR in comparison to the GS iterative method for examples 1 to 3 are summarized in Table 4.

Table 1: Comparison in terms of number of iterations.

Example	Method	Mesh Size				
		128	256	512	1024	2048
1	GS	1567	5216	16631	48516	92797
	SOR	137	261	488	931	1742
	2EGSOR	77	146	276	520	969
2	GS	53	271	1170	4649	17710
	SOR	17	36	70	139	276
	2EGSOR	14	30	61	119	234
3	GS	102	353	1295	4797	17733
	SOR	50	96	187	367	718
	2EGSOR	27	53	103	203	401

Table 2: Comparison in terms of execution time (seconds).

Example	Method	Mesh Size				
		128	256	512	1024	2048
1	GS	1.31	8.84	58.76	430.30	2907.01
	SOR	0.09	0.33	1.28	4.96	19.28
	2EGSOR	0.05	0.18	0.68	2.72	10.64
2	GS	0.05	0.28	2.30	17.75	133.89
	SOR	0.04	0.05	0.16	0.61	2.41
	2EGSOR	0.04	0.04	0.12	0.43	1.65
3	GS	0.06	0.36	2.59	19.04	140.79
	SOR	0.05	0.11	0.41	1.59	6.17
	2EGSOR	0.04	0.06	0.20	0.77	3.01

Table 3: Comparison in terms of maximum absolute error.

Example	Method	Mesh Size				
		128	256	512	1024	2048
1	GS	2.9775e-05	2.9153e-05	2.6663e-05	1.6703e-05	2.3135e-05
	SOR	2.9981e-05	2.9982e-05	2.9982e-05	2.9986e-05	2.9989e-05
	2EGSOR	2.9982e-05	2.9983e-05	2.9986e-05	3.1022e-05	2.9987e-05
2	GS	1.2755e-05	1.2752e-05	1.2700e-05	1.2456e-05	1.1348e-05
	SOR	1.2757e-05	1.2765e-05	1.2765e-05	1.2765e-05	1.2762e-05
	2EGSOR	1.2757e-05	1.2765e-05	1.2766e-05	1.2766e-05	1.2765e-05
3	GS	8.4348e-05	8.4250e-05	8.3846e-05	8.2239e-05	7.5848e-05
	SOR	8.4375e-05	8.4378e-05	8.4372e-05	8.4349e-05	8.4325e-05
	2EGSOR	8.4376e-05	8.4378e-05	8.4375e-05	8.4372e-05	8.4365e-05

Table 4: The decrement percentage for the SOR and 2EGSOR iterative methods.

Example	Method	Number of Iterations (%)	Execution Time (%)
1	SOR	91.26-98.12%	93.13-99.34%
	2EGSOR	95.09-98.96%	96.18-99.63%
2	SOR	67.92-98.44%	20.00-98.20%
	2EGSOR	73.58-98.68%	20.00-98.77%
3	SOR	50.98-95.95%	16.67-95.62%
	2EGSOR	73.53-97.74%	33.33-97.86%

5. Conclusion

In this paper, we investigate the performance of 2EGSOR iterative method in solving the one-dimensional convection-diffusion problems using the implicit finite difference schemes in second-order as in equation (3) and fourth-order as in equation (4). The numerical results depicted in Tables 1 to 3 shows that the application of the 2EGSOR iterative method has reduced the number of iterations and the execution time, better than SOR and GS iterative methods. Therefore, it can be pointed out that the developed two-point EGSOR is able to show substantial improvement in the number of iterations and execution time in comparison to the other point iterative methods. For future work, this study could be extended to investigate the performance of the proposed approximation equation using half-sweep iteration concept as discussed by Abdullah (1991), Akhir et al. (2011) and Dahalan et al. (2013).

References

Abdullah, A. R. (1991). The four point explicit decoupled group (EDG) method: A fast poisson solver. *International Journal of Computer Mathematics*, 38:61–70.

Akhir, M. K. M., Othman, M., Sulaiman, J., Majid, Z. A., and Suleiman, M. (2011). Half-sweep modified successive overrelaxation for solving iterative method two-dimensional helmholtz equations. *Australian Journal of Basic and Applied Sciences*, 5(12):3033–3039.

Dahalan, A. A., Muthuvalu, M. S., and Sulaiman, J. (2013). Numerical solutions of two-point fuzzy boundary value problem using half-sweep alternating group explicit method method two-dimensional helmholtz equations. *AIP Conference Proceedings*, 1557(1):103–107.

Dai, R., Zhang, J., and Wang, Y. (2016). Higher order ADI method with completed richardson extrapolation for solving unsteady convection-diffusion equations. *Computers Mathematics with Applications*, 71(1):431–442.

Evans, D. J. (1985). Group explicit iterative methods for solving large linear systems. *International Journal of Computer Mathematics*, 17:81–108.

Evans, D. J. and Sahimi, M. S. (1989). The alternating group explicit iterative method (AGE) to solve parabolic and hyperbolic partial differential equations. *Annual Review of Heat Transfer*, 2(2):283–389.

Evans, D. J. and Yousif, W. S. (1990). The explicit block relaxation method as a grid smoother in the multigrid V-cycle scheme. *International Journal of Computer Mathematics*, 34:71–78.

Ge, Y., Zhao, F., and Wei, J. (2018). A high order compact ADI method for solving 3D unsteady convection diffusion problems. *Applied and Computational Mathematics*, 7(1):1–10.

Hackbusch, W. (1994). *Iterative Solution of Large Sparse Systems of Equations*. Springer-Verlag, New York.

Mittal, R. C. and Jain, R. K. (2012). Redefined cubic B-splines collocation method for solving convection-diffusion equations. *Applied Mathematical Modelling*, 36(11):5555–5573.

Saad, Y. (2003). *Iterative methods for sparse linear systems*. SIAM, Philadelphia, 2nd edition.

Sulaiman, J., Hasan, M. K., Othman, M., and Karim, S. A. A. (2010). Fourth-order QSMSOR iterative method for the solution of one-dimensional parabolic PDE's. In *International Conference on Industrial and Applied Mathematics (CIAM2010)*, pages 34–39, Bandung. ITB.

Sun, H. and Li, L. (2014). A CCD-ADI method for unsteady convection-diffusion equations. *Computer Physics Communications*, 185(3):790–797.

Young, D. M. (1954). Iterative methods for solving partial difference equations of elliptic type. *Transactions of the American Mathematical Society*, 76(1):92–111.

Young, D. M. (1971). *Iterative solution of large linear systems*. Academic Press, London.

Young, D. M. (1972). Second-degree iterative methods for the solution of large linear systems. *Journal of Approximation Theory*, 5(2):137–148.

Young, D. M. (1976). *Iterative solution of linear systems arising from finite element techniques*. Academic Press, London.